Toulouse, 18th October, 2001

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

SSALTO

ALGORITHM DEFINITION, ACCURACY AND SPECIFICATION VOLUME 9 : CMA MECHANISMS

Prepared by:	J.P. DUMONT CLS P. SICARD CLS J.STUM CLS P. THIBAUT CLS O.Z. ZANIFE CLS	
Accepted by:	P. VINCENT CNES F. PARISOT CNES B.DELATTE CNES	
Approved by:	S. COUTIN-FAYE CNES	

Document ref: SMM-ST-M2-EA-11010-CN		Issue: 2	Update: 4

For application	DS2	DS4	DS5	DH2	DPP	DJ1	TP	TP2	F-PAC	DOR	AVI	SOL
								X				

Configuration controlled Document	YES	by: CCM SSALTO	Since:

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

TYPING FACILITIES

Original typing	Storage	Softwares
		Office 97

DOCUMENT STATUS

Page																		
Issue																		
Update																		

DOCUMENT CHANGE RECORD

Issue	Update	Date	Modifications	Visa
0	0	9th April, 1999	Document creation	
1	0	2nd Nov., 1999	Correction of minor errors pointed out during the software development phase, and accounting for SWT comments (Boston, June 1999) in mechanisms GEN_MEC_COM_04, GEN_MEC_COM_05, GEN_MEC_CON_02, and GEN_MEC_MOD_03. Moreover, the following mechanisms have been defined or added: GEN_MEC_GRI_03 and 04 GEN_MEC_INT_05, 06 and 07 GEN_MEC_COM_03 and 06 GEN_MEC_QUA_02, 03 and 05 GEN_MEC_CON_04 GEN_MEC_MIS_01, 02 and 03 GEN_MEC_CRO_01, 02, 03, 04, 05, 06, 07 and 08 GEN_MEC_SEL_01	
2	0	14th April, 2000	The following mechanisms have been updated: GEN_MEC_INT_03 GEN_MEC_COM_01 GEN_MEC_QUA_03 and 05 GEN_MEC_CON_04 GEN_MEC_INT_07 GEN_MEC_CRO_06, 07 and 08 GEN_MEC_SEL_01 The following mechanisms have been added: GEN_MEC_INT_08 and 09 GEN_MEC_COM_07, 08 and 09 GEN_MEC_CON_05 and 06 GEN_MEC_SEL_02 GEN_MEC_MAT_01 GEN_MEC_INT_10	

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

2	1	17th November, 2000	The following mechanisms have been updated: GEN_MEC_GRI_03 GEN_MEC_GRI_04 GEN_MEC_COM_08 GEN_MEC_COM_09 GEN_MEC_QUA_03 GEN_MEC_QUA_05 GEN_MEC_MIS_03 GEN_MEC_SEL_01 GEN_MEC_INT_07 GEN_MEC_INT_10	
2	2	9th March, 2001	The following mechanisms have been updated: GEN_MEC_GRI_02 GEN_MEC_CON_04	
2	3	4th July, 2001	The following mechanisms have been updated: GEN_MEC_COM_08 The following mechanisms have been added: GEN_MEC_INT_11, GEN_MEC_INT_12, GEN_MEC_GRI_05	
2	4	18th October, 2001	Accounting for comments from JPL (November 22, 2000), and update of GEN_MEC_INT_11	

ABBREVIATIONS

Abbreviation	Definition
ADAS	Algorithm Definition, Accuracy and Specification
ADx	Applicable Document x
CLS	Collecte Localisation Satellites
CMA	Centre Multi-missions Altimètre
CNES	Centre National d'Etudes Spatiales
FFT	Fast Fourier Transform
JPL	Jet Propulsion Laboratory
NRT	Near Real Time
OFL	Off-Line
RDx	Reference Document x
TBC	To Be Confirmed
TBD	To Be Defined

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

APPLICABLE AND REFERENCE DOCUMENTS

Reference	Document title
SMM-ST-M2-EA-11002-CN	AD1 Algorithm Definition, Accuracy and Specification Volume 1: JASON Real Time Processing
SMM-ST-M2-EA-11003-CN	AD2 Algorithm Definition, Accuracy and Specification Volume 2: CMA Altimeter Level 1b Processing
SMM-ST-M2-EA-11004-CN	AD3 Algorithm Definition, Accuracy and Specification Volume 3: CMA Radiometer Level 1b Processing
SMM-ST-M2-EA-11005-CN	AD4 Algorithm Definition, Accuracy and Specification Volume 4: CMA Altimeter Level 2 Processing
SMM-ST-M2-EA-11006-CN	AD5 Algorithm Definition, Accuracy and Specification Volume 5: CMA Radiometer Level 2 Processing
SMM-ST-M2-EA-11007-CN	AD6 Algorithm Definition, Accuracy and Specification Volume 6: CMA Altimeter/Radiometer Verification processing
SMM-ST-M2-EA-11008-CN	AD7 Algorithm Definition, Accuracy and Specification Volume 7: CMA Real Time Processing Control
SMM-ST-M2-EA-11009-CN	AD8 Algorithm Definition, Accuracy and Specification Volume 8: CMA Off-Line Processing Control
SMM-SP-M2-EA-32012-CLS	RD1 CMA production: Specifications of the Data management Algorithms
Numerical Recipes, Ed. 2	RD2 The Art of Scientific Computing in C (Edition 2). William H. Press, Brian P. Flaneery, Saul A. Teukolsky, William T. Vetterling
	RD3 NAG Fortran Library Manual – Mark 18

TBC AND TBD LIST

TBC/TBD	Section	Brief description
/	/REFREF	/

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

CONTENTS

1. INTRODUCTION 1
2. MECHANISMS 2
GEN_MEC_GRI_01 - Cell identification 3
GEN_MEC_GRI_02 - Meteo cell identification 7
GEN_MEC_GRI_03 - Calculation of a grid of a parameter 12
GEN_MEC_GRI_04 - Smoothing of a grid of a parameter 16
GEN_MEC_GRI_05 - Calculation of a non regular grid of a parameter 22
GEN_MEC_INT_01 - Linear weighting 26
GEN_MEC_INT_02 - Linear interpolation 29
GEN_MEC_INT_03 - Bilinear interpolation 31
GEN_MEC_INT_04 - Bivariate interpolation from six points 34
GEN_MEC_INT_05 - Interpolation of an array at a new sampling frequency 37
GEN_MEC_INT_06 - Everett interpolation 40
GEN_MEC_INT_07 - Spline interpolation 43
GEN_MEC_INT_08 - One-dimensional polynomial interpolation 46
GEN_MEC_INT_09 - Computation of the coefficients of a polynomial interpolation on a two dimension grid 50
GEN_MEC_INT_10 - Interpolation of altimeter measurements at a given point along-track 55
GEN_MEC_INT_11 - Interpolation of 20-Hz altimeter measurements at a given point along-track 59
GEN_MEC_INT_12 - Spline interpolation of grid values 64
GEN_MEC_COM_01 - Arithmetic averaging 68
GEN_MEC_COM_02 - Linear regression / Least square method 71
GEN_MEC_COM_03 - Linear regression / Absolute deviation method 74
GEN_MEC_COM_04 - Editing and compression 77
GEN_MEC_COM_05 - Estimation of the slope of the logarithm of the trailing edge of a waveform (linear regression) 81
GEN_MEC_COM_06 - Flag compression 84
GEN_MEC_COM_07 - Polynomial regression / Least Square method 86
GEN_MEC_COM_08 - Least-squares fit by polynomial or periodic functions 90
GEN_MEC_COM_09 - Multiple linear regression 94
GEN_MEC_QUA_01 - Quality check from thresholds 98
GEN_MEC_QUA_03 - Editing of measurements against thresholds 100
GEN_MEC_QUA_04 - Check of a parameter versus a reference value 104
GEN_MEC_QUA_05 - Editing of measurements using spline 107
GEN_MEC_COR_01 - Computation of a tracking/retracking combined parameter 112
GEN_MEC_COR_02 - Computation of a corrected parameter from its raw value and its corrections 115
GEN_MEC_COR_03 - Computation of the Doppler correction on the altimeter range 117
GEN_MEC_COR_04 - Computation of a modeled instrumental correction 119
GEN_MEC_COR_05 - Calculation of a path delay value 123
GEN_MEC_CON_01 - Derivation of the significant waveheight from the composite Sigma 127
GEN_MEC_CON_02 - Derivation of the square of the off-nadir angle from the slope of the trailing edge 130
GEN_MEC_CON_03 - Conversion of a position vector from geodetic to Cartesian co-ordinates 133
GEN_MEC_CON_04 - Time conversions 136
GEN_MEC_CON_05 - Conversion between geographic and geomagnetic latitude 143
GEN_MEC_CON_06 - Conversion of a position vector from Cartesian to geodetic co-ordinates 147
GEN_MEC_MOD_01 - Ocean echo model and partial derivatives: Initialization (1) 151
GEN_MEC_MOD_02 - Ocean echo model and partial derivatives: Initialization (2) 154
GEN_MEC_MOD_03 - Ocean echo model and partial derivatives: Computation 157
GEN_MEC_MOD_04 - Computation of "Gamma" and "Alpha" altimeter parameters 164
GEN_MEC_MIS_01 - Ground distance between two points 166
GEN_MEC_MIS_02 - Search of the closest corresponding altimeter point in a reference track 169
GEN_MEC_MIS_03 - Location of missing measurements 173
GEN_MEC_CRO_01 - Search of two consecutive measurements 178
GEN_MEC_CRO_02 - Presence of a common latitude domain 181
GEN_MEC_CRO_03 - Presence of a common longitude domain 183
GEN_MEC_CRO_04 - Bisection 186
GEN_MEC_CRO_05 - Change of the sign of the latitude difference 189
GEN_MEC_CRO_06 - Search of one boundary of the common longitude domain 191
GEN_MEC_CRO_07 - Search on one arc of the closest longitude smaller than a given longitude of the other arc 201
GEN_MEC_CRO_08 - Search of the intersecting segments 208
GEN_MEC_SEL_01 - Selection of N points for spline calculation 213
GEN_MEC_SEL_02 - Selection of the satellite longitude and local time for a given latitude 217
GEN_MEC_MAT_01 - Updating a QR decomposition using Givens rotation 221

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

1. INTRODUCTION

The definition of the various procedures of the CMA and the specification of the "scientific" algorithms (i.e. the cores of these procedures) are given in the "Algorithm Definition Accuracy and Specification" documents:

- AD1 to AD5 for the "production" algorithms
- AD7 and AD8 for the "control" algorithms
- AD6 for the "expertise" algorithms

The specification of the "data management" algorithms, which represent the algorithms aimed at preparing the data requested for the scientific algorithms is given in RD1.

This document is a complement to the above-mentioned documents, and provides the specification of the functions common to several algorithms (e.g. averaging) or the functions frequently requested within an algorithm (e.g. linear interpolation).

Organization of the document

Each algorithm (mechanism) is described, using the following items:

- Name and identifier of the algorithm
- Function
- Algorithm specification:
  - Input data
  - Output data
  - Processing
- Comments (if any)
- References (if any)

Basic rules

The following basic rules are applied to the specification of the algorithms:

- The specifications of an algorithm are always relevant to the processing of a single data point and not to a set of data points
- The input and output data are always identified by a precise description, an explicit name (that could be used in the coding phase), a unit and if necessary a reference system
- Regarding the errors that may occur during the processing functions (for example, negative argument for logarithmic or square root functions), the algorithms systematically output an execution status. The building and the management of this information will be defined during the architectural design of the software.
- Regarding the representation of tables, the following conventions are used in the following:
  - $X[N_1:N_2]$ represents a one-dimension table whose elements are $X(i)$ (or X_i) with $i \in [N_1, N_2]$

$X[N_1:N_2][M_1:M_2]$ represents a two-dimension table whose elements are $X(i,j)$ (or X_{ij}) with $i \in [N_1,N_2]$ and $j \in [M_1,M_2]$, and so on

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

2. MECHANISMS

In the following specifications:

- INT(x) represents the function which returns the integer part of x
- CEIL(x) represents the function which returns the smallest integer value greater than or equal to x
- FLOOR(x) represents the function which returns the largest integer value less than or equal to x
- NINT(x) represents the function which rounds x to the nearest integer value

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_GRI_01 - Cell identification

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	P. SICARD CLS	
Checked by:	For the JASON-1 SWT C.K. SHUM For the JASON-1 Project P. VINCENT CNES S. DESAI JPL For the F-PAC / ENVISAT Project J. BENVENISTE ESA P. VINCENT CNES	
Approved by:	P. VINCENT CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To localize the four table points (or grid points) surrounding the (X,Y) point and to compute the weight of these four points.

ALGORITHM SPECIFICATION

Input data

- X co-ordinate of the point : X
- Y co-ordinate of the point : Y
- Table step (or grid step) in X [(1)] : DX
- Table step (or grid step) in Y [(2)] : DY
- Value of X corresponding to the first table point (or grid point) : Xfir
- Value of Y corresponding to the first table point (or grid point) : Yfir
- Number of points in X of the table (or the grid) : Nb_PtX
- Number of points in Y of the table (or the grid) : Nb_PtY
- Cycling value for the X variable : Xcyc (≥ 0)
- Cycling value for the Y variable : Ycyc (≥ 0)
- Truncation flag for the X variable : Xcut {0,1}
- Truncation flag for the Y variable : Ycut {0,1}

Output data

- Indexes of the four table points (or grid points) surrounding the (X,Y) point:
  - Left index in X : Ileft (/)
  - Right index in X : Iright (/)
  - Bottom index in Y : Jbottom (/)
  - Top index in Y : Jtop (/)
- Weight of these four points:
  - Weight of the lower left corner : Wll (/)
  - Weight of the lower right corner : Wlr (/)
  - Weight of the upper left corner : Wul (/)
  - Weight of the upper right corner : Wur (/)

---

(1) DX is always positive. In the case of a geographic grid, the grid always goes from West to East.

(2) In the case of a geographic grid, DY is positive if the grid goes from South to North, and negative otherwise.

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- Execution status

Processing

- Computing the values of Xlas and Ylas corresponding to the last value of the table points or grid points:
  - Xlas = Xfir + (Nb_PtX – 1) ∗DX (1)
  - If (DY > 0) then Yinf = Yfir; Ylas = Yinf + (Nb_PtY –1)∗DY (2)
  - If (DY < 0) then Ylas = Yfir; Yinf = Ylas + (Nb_PtY –1) ∗DY (3)
- Checking the (X,Y) value relative to the boundaries and changing their values according to the cycling and truncation flags:
  - If (Xcyc = 0) and (X < Xfir), then: (4)
    - If (Xcut = 1), then X = Xfir
    - If (Xcut = 0), then set Execution Status to “Point (X,Y) is outside the model” and exit
  - If (Xcyc = 0) and (X > Xlas), then: (5)
    - If (Xcut = 1), then X = Xlas
    - If (Xcut = 0), then set Execution Status to “Point (X,Y) is outside the model” and exit
  - If (Xcyc > 0) and (X < Xfir), then X = X + Xcyc (6)
  - If (Ycyc = 0) and (Y < Yinf), then: (7)
    - If (Ycut = 1), then Y = Yinf
    - If (Ycut = 0), then set Execution Status to “Point (X,Y) is outside the model” and exit
  - If (Ycyc = 0) and (Y > Ylas), then: (8)
    - If (Ycut = 1), then Y = Ylas
    - If (Ycut = 0), then set Execution Status to “Point (X,Y) is outside the model” and exit
  - If (Ycyc > 0) and (Y < Yinf), then Y = Y + Ycyc (9)
- Computing the indexes (Ileft,Jbottom) of the lower left corner of the cell surrounding the (X,Y) point:
  - $\text{Ileft} = \text{INT}\left(\frac{X - \text{Xfir}}{\text{DX}}\right)$ (10)
  - If (DY > 0) then $\text{Jbottom} = \text{FLOOR}\left(\frac{Y - \text{Yfir}}{\text{DY}}\right)$ (11)
  - If (DY < 0) then $\text{Jbottom} = \text{CEIL}\left(\frac{Y - \text{Yfir}}{\text{DY}}\right)$
- Computing the indexes (Iright,Jtop) of the upper right corner of the cell surrounding the (X,Y) points:
  - Iright = Ileft + 1 (12)

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- If (Iright ≥ Nb_PtX), then: (13)
  - If (Xcyc > 0),then Iright = Iright - Nb_PtX
  - If (Xcyc = 0), then Iright = Ileft
- If (DY > 0), then: (14)
  - Jtop = Jbottom + 1 (15)
  - If (Jtop ≥ Nb_PtY), then: (16)
    - If (Ycyc > 0), then Jtop = Jtop – Nb_PtY
    - If (Ycyc = 0), then Jtop = Jbottom

  Else (DY < 0): (17)
  - Jtop = Jbottom - 1 (18)
  - If (Jtop < 0), then: (19)
    - If (Ycyc > 0), then Jtop = Jtop + Nb_PtY
    - If (Ycyc = 0), then Jtop = Jbottom

- Computing the distance A, within the cell, of the X abscissa relative to the (Ileft, Jbottom) point, normalized to the cell length DX , and the distance B, within the cell, of the Y ordinate relative to the (Ileft, Jbottom) point, normalized to the cell length DY:

$$A = \frac{X - (Xfir + Ileft * DX)}{DX} \quad (20)$$

$$B = \frac{Y - (Yfir + Jbottom * DY))}{|DY|} \quad (21)$$

- Computing the relative weight of each cell point for the bilinear interpolation: this weight is equal to the area of the cell portion delimited by the measurement point and the opposite cell point:

Wll = (1-A) ∗ (1-B) (Lower Left corner) (22)

Wlr = A ∗ (1-B) (Lower Right corner) (23)

Wul = (1-A) ∗ B (Upper Left corner) (24)

Wur = A ∗ B (Upper Right corner) (25)

COMMENTS

None

REFERENCES

None

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_GRI_02 - Meteo cell identification

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J. STUM CLS P. SICARD CLS	
Checked by:	For the JASON-1 SWT L. EYMARD For the JASON-1 Project P. VINCENT CNES S. DESAI JPL For the F-PAC / ENVISAT Project J. BENVENISTE ESA P. VINCENT CNES	
Approved by:	P. VINCENT CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	Date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To identify in the ECMWF model non-regular grid the four grid points surrounding the measurement point and to compute the weight of these four points for the bilinear interpolation.

ALGORITHM SPECIFICATION

Input data

- Latitude of the measurement point : Lat_Mean (degrees)
- Longitude of the measurement point : Lon_Mean (degrees)
- Longitude of the first grid point : Lon_First (degrees)
- Number of grid points in latitude : Nb_Lat (/)
- Table providing the latitudes of the model grid points : Tab_Lat[0:Nb_Lat-1] (deg)
- Table providing the number of grid points in longitude for each latitude of the model : Tab_Nb_Lon[0:Nb_Lat-1] (/)

Output data

- The cell vector, C[0:3], containing the indexes of the four meteorological vector elements which define the cell of 4 grid points surrounding the measurement point.
- The weight vector, W[0:3], containing the weights associated with these four points.
- Execution status

Processing

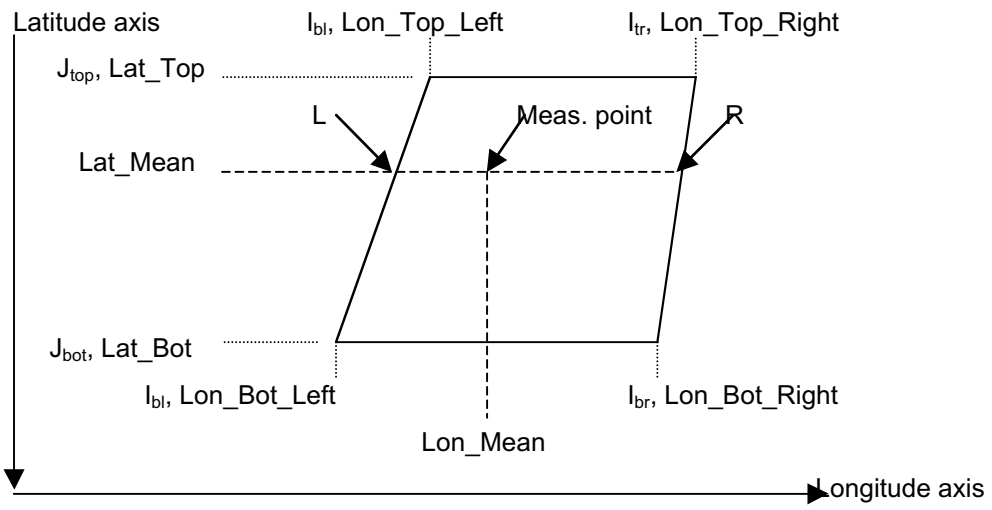


Figure 1

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Check the input value of measurement latitude relative to the grid limits:

If [(Lat_Mean > Tab_Lat[0]) OR (Lat_Mean < Tab_Lat[Nb_Lat – 1)], then:

- The status is set to “latitude outside the model grid”
- Exit

- Inverting the model grid points latitudes to get an increasing series :

  For I = 0 to Nb_Lat – 1

  - Lat_Used[I] = Tab_Lat[Nb_Lat – 1 – I] (1)

- Computing the index J_{used} in Lat_Used corresponding to the latitude of the bottom point of the cell surrounding the measurement point, using the mechanism “GEN_MEC_CRO_04 – Bisection”,

  The input parameters of which are :

  - Nbmes = Nb_Lat
  - Lon[0:Nbmes – 1] = Lat_Used[0:Nb_Lat – 1]
  - Lon_Giv = Lat_Mean
  - I_Min = 0
  - I_Max = Nb_Lat – 1

  And the output parameters of which are :

  - I_Clos = J_{used}
  - The execution status

- Computing the corresponding index J_{bot} in Tab_Lat :

  J_{bot} = Nb_Lat – 1 – J_{used} (2)

- Computing the index J_{top} in Tab_Lat corresponding to the latitude of the top point of the cell surrounding the measurement point :

  $J_{top} = J_{bot} - 1$ (3)

- Computing the latitudes of the bottom and top points of the cell surrounding the measurement point:

  Lat_Bot = Tab_Lat[J_{bot}] (4)

  Lat_Top = Tab_Lat[J_{top}] (5)

- Computing the indexes (relative to the first longitude) and the longitudes of the bottom left, bottom right, top left and top right grid points of the cell surrounding the measurement point:

  - If Lon_Mean < Lon_First, then: Lon_Mean = Lon_Mean + 360. (6)
  - Bottom left index and longitude:

    Nb_Lon = Tab_Nb_Lon(J_{bot}) (7)

    Lon_Step = 360. / Nb_Lon (8)

    $$I_{bl} = \mathrm{INT}\left(\frac{\mathrm{Lon_Mean} - \mathrm{Lon_First}}{\mathrm{Lon_Step}}\right) \quad (9)$$

    Lon_Bot_Left = Lon_First + $I_{bl} * $ Lon_Step (10)

  Bottom right index and longitude:

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

$I_{br} = I_{bl} + 1$ (11)

Lon_Bot_Right = Lon_First + I_{br} ∗ Lon_Step (12)

If $I_{br} \geq$ Nb_Lon, then: $I_{br} = I_{br}$ – Nb_lon (13)

– Top left index and longitude:

Nb_Lon = Tab_Nb_Lon(J_{top}) (14)

Lon_Step = 360. / Nb_Lon (15)

$$I_{tl} = \mathrm{INT}\left(\frac{\mathrm{Lon_Mean} - \mathrm{Lon_First}}{\mathrm{Lon_Step}}\right) \quad (16)$$

Lon_Top_Left = Lon_First + I_{tl} ∗ Lon_Step (17)

– Top right index and longitude:

$I_{tr} = I_{tl} + 1$ (18)

Lon_Top_Right = Lon_First + I_{tr} ∗ Lon_Step (19)

If $I_{tr} \geq$ Nb_Lon, then: $I_{tr} = I_{tr}$ – Nb_lon (20)

- Computing the absolute indexes of the bottom left, bottom right, top left and top right grid points of the cell surrounding the measurement point:

$$K_{bl} = I_{bl} + \sum_{L=0}^{J_{bot}-1} \mathrm{Tab_Nb_Lon}(L) \quad (21)$$

$$K_{br} = I_{br} + \sum_{L=0}^{J_{bot}-1} \mathrm{Tab_Nb_Lon}(L) \quad (22)$$

$$K_{tl} = I_{tl} + \sum_{L=0}^{J_{top}-1} \mathrm{Tab_Nb_Lon}(L) \quad (23)$$

$$K_{tr} = I_{tr} + \sum_{L=0}^{J_{top}-1} \mathrm{Tab_Nb_Lon}(L) \quad (24)$$

- Computing the longitude of point L (see **Figure 1**), Lon_L:

$$\mathrm{Lon_L} = \mathrm{Lon_Top_Left} + (\mathrm{Lon_Bot_Left} - \mathrm{Lon_Top_Left}) * \frac{\mathrm{Lat_Top} - \mathrm{Lat_Mean}}{\mathrm{ABS}(\mathrm{Lat_Top} - \mathrm{Lat_Bot})} \quad (25)$$

- Computing the longitude of point R (see **Figure 1**), Lon_R:

$$\mathrm{Lon_R} = \mathrm{Lon_Top_Right} + (\mathrm{Lon_Bot_Right} - \mathrm{Lon_Top_Right}) * \frac{\mathrm{Lat_Top} - \mathrm{Lat_Mean}}{\mathrm{ABS}(\mathrm{Lat_Top} - \mathrm{Lat_Bot})} \quad (26)$$

Computing the distance A, within the cell, of the measurement point longitude relative to the longitude of point L, normalized to the longitude step (Lon_R – Lon_L):

$$A = \frac{\mathrm{Lon_Mean} - \mathrm{Lon_L}}{\mathrm{Lon_R} - \mathrm{Lon_L}} \quad (27)$$

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- Computing the distance B, within the cell, of the measurement point latitude relative to the cell bottom latitude, normalized to the latitude step:

$$B = \frac{Lat_Mean - Lat_Bot}{ABS(Lat_Top - Lat_Bot)} \quad (28)$$

- Computing the relative weight of each cell point for the bilinear interpolation:
  - weight of bottom left cell point : $W_{bl} = (1 - A) * (1 - B)$ (29)
  - weight of bottom right cell point : $W_{br} = A * (1 - B)$ (30)
  - weight of top left cell point : $W_{tl} = (1 - A) * B$ (31)
  - weight of top right cell point : $W_{tr} = A * B$ (32)

- Defining the cell vector, C, and the weight vector, W:

$$C(0) = K_{bl} \quad (33)$$

$$C(1) = K_{br} \quad (34)$$

$$C(2) = K_{tl} \quad (35)$$

$$C(3) = K_{tr} \quad (36)$$

$$W(0) = W_{bl} \quad (37)$$

$$W(1) = W_{br} \quad (38)$$

$$W(2) = W_{tl} \quad (39)$$

$$W(3) = W_{tr} \quad (40)$$

COMMENT

The number of grid points in longitude is a function of latitude. For the version T213/L31 of the ECMWF model (320 latitudes), this number runs from 18 at the poles to 640 at the equator.

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_GRI_03 - Calculation of a grid of a parameter

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J. STUM	CLS	
Checked by:	N. PICOT	CNES	
Approved by:	P. VINCENT	CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To compute the averaged value of a parameter on each cell of a latitude-longitude regular grid.

ALGORITHM SPECIFICATION

Input data

- Number of measurement points : Nb_Pts (/)
- Latitudes of the measurement points : Lat_Pts[0:Nb_Pts-1] (degree)
- Longitudes of the measurement points : Lon_Pts[0:Nb_Pts-1] (degree)
- Values of the parameter : Val_Pts[0:Nb_Pts-1]
- Number of grid points in longitude : Nb_Grid_Lon (/)
- Minimum longitude : Lon_Min (degree)
- Maximum longitude : Lon_Max (degree)
- Longitude step : Lon_Step (degree)
- Number of grid points in latitude : Nb_Grid_Lat (/)
- Minimum latitude : Lat_Min (degree)
- Maximum latitude : Lat_Max (degree)
- Latitude step : Lat_Step (degree)
- Minimum threshold for the parameter value : Threshold_Min
- Maximum threshold for the parameter value : Threshold_Max

Output data

- Grid_Nb_Points[0:Nb_Grid_Lon-1][0:Nb_Grid_Lat-1] (grid of the number of points in each cell)
- Grid_Mini[0:Nb_Grid_Lon-1][0:Nb_Grid_Lat-1] (grid of the minimum of the parameter value in each cell)
- Grid_Maxi[0:Nb_Grid_Lon-1][0:Nb_Grid_Lat-1] (grid of the maximum of the parameter value in each cell)
- Grid_Mean[0:Nb_Grid_Lon-1][0:Nb_Grid_Lat-1] (grid of the mean of the parameter value in each cell)
- Grid_Std[0:Nb_Grid_Lon-1][0:Nb_Grid_Lat-1] (grid of the standard deviation of the parameter value in each cell)
- Execution status

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Processing

- For I_Lon = 0 to Nb_Grid_Lon-1:
  - For I_Lat = 0 to Nb_Grid_Lat-1:

    Nb_Pts_Per_Cell[I_Lon][I_Lat] = 0 (1)
- If Lon_Min > Lon_Max, then:

  Lon_Min = Lon_Min – 360 (2)
- For each input measurement point (for I_Mes = 0 to Nb_Pts-1):
  - If Lon_Min < 0, then:
    - If (Lon_Pts[I_Mes] > Lon_Max then:
      - Lon_Pts[I_Mes] = Lon_Pts[I_Mes] – 360 (3)

  Calculation of the grid cell co-ordinates in the grid:

  - $I_Lon = INT(\frac{Lon_Pts[I_Mes] - Lon_Min}{Lon_Step})$ (4)
  - $J_Lat = INT(\frac{Lat_Pts[I_Mes] - Lat_Min}{Lat_Step})$ (5)
  - Val_Flag = 0 (6)

  Checking if the measurement point is to be included in the averaging process, according to its location and value:

  - If ((I_Lon < 0) OR (I_Lon > Nb_Grid_Lon – 1)) then:
    - Val_Flag = 1 (7)
  - If ((J_Lat < 0) OR (J_Lat > Nb_Grid_Lat – 1)) then:
    - Val_Flag = 1 (8)
  - If Threshold_Min ≠ Threshold_Max (i.e., if selection on the parameter values is required):
    - If ((Val_Pts[I_Mes] < Threshold_Min) OR (Val_Pts[I_Mes] > Threshold_Max)) then:
      - Val_Flag = 1 (9)
  - If Val_Flag = 0, then :

    Distribution of the parameter value in its cell :

    - Val_Pts_Per_Cell[I_Lon][I_Lat][Nb_Pts_Per_Cell[I_Lon][I_Lat]] = Val_Pts[I_Mes] (10)
    - Nb_Pts_Per_Cell[I_Lon][I_Lat] = Nb_Pts_Per_Cell[I_Lon][I_Lat] + 1 (11)

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- For I_Lon = 0 to Nb_Grid_Lon-1
  - For I_Lat = 0 to Nb_Grid_Lat-1
    - Grid_Nb_Points[I_Lon][I_Lat] = Nb_Pts_Per_Cell[I_Lon][I_Lat] (12)
    - The minimum value, maximum value, mean value and standard deviation of the parameter is computed in each cell, using the mechanism “GEN_MEC_COM_01 - Arithmetic averaging”

      The input parameters of which are :

      ◊ N = Nb_Pts_Per_Cell[I_Lon][I_Lat]

      ◊ Val[0:N-1] = Val_Pts_Per_Cell[I_Lon][I_Lat][0:Nb_Pts_Per_Cell[I_Lon][I_Lat]-1]

      And the output parameters of which are :

      ◊ Mean = Grid_Mean[I_Lon][I_Lat]

      ◊ Std = Grid_Std[I_Lon][I_Lat]

      ◊ Min = Grid_Mini[I_Lon][I_Lat]

      ◊ Max = Grid_Maxi[I_Lon][I_Lat]

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_GRI_04 - Smoothing of a grid of a parameter

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J. STUM	CLS	
Checked by:	N. PICOT	CNES	
Approved by:	P. VINCENT	CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To smooth the mean cell value of a parameter of a latitude-longitude regular grid. The smoothed cell value is a weighted mean of the neighboring cell values. Weighting is inversely to the distance between the neighbor and the cell.

ALGORITHM SPECIFICATION

Input data

- Grid of the mean value of the parameter : Mean[0:Nb_Grid_Lon-1][0:Nb_Grid_Lat-1]
- Number of points per cell of the parameter : Nb_Pts[0:Nb_Grid_Lon-1][0:Nb_Grid_Lat-1]
- Characteristics of the grid :
  - Number of grid points in longitude : Nb_Grid_Lon (/)
  - Minimum longitude : Lon_Min (degree)
  - Maximum longitude : Lon_Max (degree)
  - Longitude step : Lon_Step (degree)
  - Number of grid points in latitude : Nb_Grid_Lat (/)
  - Minimum latitude : Lat_Min (degree)
  - Maximum latitude : Lat_Max (degree)
  - Latitude step : Lat_Step (degree)
- Number of neighboring cells in longitude to consider for the smoothing : Nb_Cells_Lon (/)
- Number of neighboring cells in latitude to consider for the smoothing : Nb_Cells_Lat (/)

Output data

- Characteristics of the smoothed grid :
  - Number of grid points in longitude : Nb_Grid_Lon_S (/)
  - Minimum longitude : Lon_Min_S (degree)
  - Maximum longitude : Lon_Max_S (degree)
  - Longitude step : Lon_Step_S (degree)
  - Number of grid points in latitude : Nb_Grid_Lat_S (/)
  - Minimum latitude : Lat_Min_S (degree)
  - Maximum latitude : Lat_Max_S (degree)
  - Latitude step : Lat_Step_S (degree)
- Grid of the smoothed mean value of the parameter :Sm_Mean[0:Nb_Grid_Lon_S-1][0:Nb_Grid_Lat_S-1]
- Execution status

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Processing

From the input grid, a new grid (« smoothed » grid) is built by oversampling the input grid, as follows (for a worldsize input grid, this leads to create a new grid with 0.25°x0.25° cells):

- Lat_Step_S = 0.1 (1)
- Nb_Grid_Lat_S = (Lat_Max + Lat_Step – Lat_Min)/Lat_Step_S (2)
- Do while (Nb_Grid_Lat_S > 720)
  - Lat_Step_S = Lat_Step_S + 0.05 (3)
  - Nb_Grid_Lat_S = (Lat_Max + Lat_Step – Lat_Min)/Lat_Step_S (4)
- Lon_Step_S = 0.1 (5)
- Nb_Grid_Lon_S = (Lon_Max + Lon_Step – Lon_Min)/Lon_Step_S (6)
- Do while (Nb_Grid_Lon_S > 1440)
  - Lon_Step_S = Lon_Step_S + 0.05 (7)
  - Nb_Grid_Lon_S = (Lon_Max + Lon_Step – Lon_Min)/Lon_Step_S (8)
- For I_Lon_S = 0 to Nb_Grid_Lon_S – 1
  - Lon_S = Lon_Min + I_Lon_S∗Lon_Step_S (9)
  - I_Lon = (Lon_S – Lon_Min)/Lon_Step (10)
  - For J_Lat_S = 0 to Nb_Grid_Lat_S – 1
    - Lat_S = Lat_Min + J_Lat_S∗Lat_Step_S (11)
    - I_Lat= (Lat_S – Lat_Min)/Lat_Step (12)
    - If Nb_Pts[I_Lon][I_Lat] > 0, then :
      - Mean_S[I_Lon_S][J_Lat_S] = Mean[I_Lon][J_Lat] (13)
    - Else
      - Mean_S[I_Lon_S][J_Lat_S] is set to its default value (14)

Then, the smoothed cell mean values are computed by averaging the individual mean values reported in the neighboring cell values. Weighting is applied in the averaging process, inversely to the distance between the neighbor and the cell.

- For I_Lon_S = 0 to Nb_Grid_Lon_S - 1
  - I_Lon_Beg = I_Lon_S – Nb_Cells_Lon (15)
  - If I_Lon_Beg < 0, then :
    - If (Lon_Max + Lon_Step – Lon_Min) = 360, then :
      - I_Lon_Beg = I_Lon_Beg + Nb_Grid_Lon_S (16)
    - Else :
      - I_Lon_Beg = 0 (17)
  - I_Lon_End = I_Lon_S + Nb_Cells_Lon (18)

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- If I_Lon_End > Nb_Grid_Lon_S – 1, then
  - If (Lon_Max + Lon_Step – Lon_Min) = 360, then :
    - I_Lon_End = I_Lon_End – Nb_Grid_Lon_S (19)
  - Else :
    - I_Lon_End = Nb_Grid_Lon_S – 1 (20)
- For I_Lat_S = 0 to Nb_Grid_Lat_S-1

  If Mean_S[I_Lon_S][I_Lat_S] is not set to default, then :
  - I_Lat_Beg = I_Lat_S – Nb_Cells_Lat (21)
  - If I_Lat_Beg < 0, then :
    - I_Lat_Beg = 0 (22)
  - I_Lat_End = I_Lat_S + Nb_Cells_Lat (23)
  - If I_Lat_End > Nb_Grid_Lat_S – 1, then :
    - I_Lat_End = Nb_Grid_Lat_S – 1 (24)
  - Nb_Cells = 0 (25)
  - If I_Lon_Beg < I_Lon_End, then :
    - For K_Lon_S = I_Lon_Beg to I_Lon_End
      - For K_Lat_S = I_Lat_Beg to I_Lat_End
        - If Mean_S[K_Lon_S][K_Lat_S] is not set to its default value, then :
          - Nb_Cells = Nb_Cells + 1 (26)
          - Dist_Lon = I_Lon_S – K_Lon_S (27)
          - Dist_Lat = I_Lat_S – K_Lat_S (28)
          - $Dist2 = Dist_Lon^2 + Dist_Lat^2$ (29)
          - If Dist2 > 0, then :

$$Weight[Nb_Cells-1] = \frac{1}{\sqrt{Dist2}} \quad (30)$$

          - Else :

$$Weight[Nb_Cells-1] = 1 \quad (31)$$

          - Value[Nb_Cells-1] = Mean_S[K_Lon_S][K_Lat_S] (32)
  - Else :
    - For K_Lon_S = I_Lon_Beg to Nb_Grid_Lon_S – 1
      - For K_Lat_S = I_Lat_Beg to I_Lat_End
        - If Mean_S[K_Lon_S][K_Lat_S] is not set to its default value, then :
          - Nb_Cells = Nb_Cells + 1 (33)

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

! If ABS(I_Lon_S – K_Lon_S) > Nb_Cells_Lon, then

$$Dist_Lon = Nb_Grid_Lon_S - ABS(I_Lon_S - K_Lon_S) \quad (34)$$

! Else :

$$Dist_Lon = I_Lon_S - K_Lon_S \quad (35)$$

! $Dist_Lat = I_Lat_S - K_Lat_S$ (36)

! $Dist2 = Dist_Lon^2 + Dist_Lat^2$ (37)

! If Dist2 > 0, then :

$$Weight[Nb_Cells-1] = \frac{1}{\sqrt{Dist2}} \quad (38)$$

! Else :

$$Weight[Nb_Cells-1] = 1 \quad (39)$$

! $Value[Nb_Cells-1] = Mean_S[K_Lon_S][K_Lat_S]$ (40)

◊ For K_Lon_S = 0 to I_Lon_End

⇒ For K_Lat_S = I_Lat_Beg to I_Lat_End

→ If Mean_S[K_Lon_S][K_Lat_S] is not set to its default value, then :

! $Nb_Cells = Nb_Cells + 1$ (41)

! If ABS(I_Lon_S – K_Lon_S) > Nb_Cells_Lon, then

$$Dist_Lon = Nb_Grid_Lon_S - ABS(I_Lon_S - K_Lon_S) \quad (42)$$

! Else :

$$Dist_Lon = I_Lon_S - K_Lon_S \quad (43)$$

! $Dist_Lat = I_Lat_S - K_Lat_S$ (44)

! $Dist2 = Dist_Lon^2 + Dist_Lat^2$ (45)

! If Dist2 > 0, then :

$$Weight[Nb_Cells-1] = \frac{1}{\sqrt{Dist2}} \quad (46)$$

! Else :

$$Weight[Nb_Cells-1] = 1 \quad (47)$$

! $Value[Nb_Cells-1] = Mean_S[K_Lon_S][K_Lat_S]$ (48)

∗ If Nb_Cells > 0, then :

◊ $Sum_Values = 0$ (49)

◊ $Sum_Weights = 0$ (50)

◊ For J = 0 to Nb_Cells – 1

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

⇒ Sum_Values = Sum_Values + Value[J] x Weight[J] (51)

⇒ Sum_Weights = Sum_Weights + Weight[J] (52)

◊ Sm_Mean[I_Lon_S][I_Lat_S] = Sum_Values/Sum_Weights (53)

Else (Mean_S[I_Lon_S][J_Lat_S] set to default), then Sm_Mean[I_Lon_S][I_Lat_S] is set to default

- Lon_Min_S = Lon_Min (54)
- Lon_Max_S = Lon_Min + (Nb_Grid_Lon_S – 1)∗ Lon_Step_S (55)
- Lat_Min_S = Lat_Min (56)
- Lat_Max_S = Lat_Min + (Nb_Grid_Lat_S – 1)∗ Lat_Step_S (57)

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_GRI_05 - Calculation of a non regular grid of a parameter

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J. STUM	CLS	
Checked by:	N. PICOT	CNES	
Approved by:	P. VINCENT	CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To compute the averaged value of a parameter on each cell of a non regular grid (the number of grid cells in longitude is function of latitude)

ALGORITHM SPECIFICATION

Input data

- Number of measurement points : Nb_Pts (/)
- Latitudes of the measurement points : Lat_Pts[0:Nb_Pts-1] (degree)
- Longitudes of the measurement points : Lon_Pts[0:Nb_Pts-1] (degree)
- Values of the parameter : Val_Pts[0:Nb_Pts-1]
- Distance between two grid points in latitude : Dist_Lat_Step (m)
- Minimum latitude of the grid : Lat_Min (degrees)
- Number of grid points in latitude : Nb_Lat (/)
- Table providing the number of grid points in longitude for each latitude of the grid : Tab_Nb_Lon[0:Nb_Lat-1] (/)
- Distance between two grid points in longitude : Dist_Lon_Step (m)
- Minimum longitude of the grid (the same for all latitudes) : Lon_Min (degrees)
- Number of grid points : Nb_Cells (/)
- Minimum threshold for the parameter value : Threshold_Min
- Maximum threshold for the parameter value : Threshold_Max

Output data

- Grid_Nb_Points[0:Nb_Cells-1] (grid of the number of points in each cell)
- Grid_Mini[0:Nb_Cells-1] (grid of the minimum of the parameter value in each cell)
- Grid_Maxi[0:Nb_Cells-1] (grid of the maximum of the parameter value in each cell)
- Grid_Mean[0:Nb_Cells-1] (grid of the mean of the parameter value in each cell)
- Grid_Std[0:Nb_Cells-1] (grid of the standard deviation of the parameter value in each cell)
- Execution status

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Processing

- For I_Cell = 0 to Nb_Cells-1:
  - Nb_Pts_Per_Cell[I_Cell] = 0 (1)
- $\text{Lat_Step} = \frac{\text{Dist_Lat_Step}}{111000}$ (2)
- For each input measurement point (for I_Mes = 0 to Nb_Pts-1):

  Calculation of the latitude index of the grid cell :

  - $\text{J_Lat} = \text{INT}\left(\frac{\text{Lat_Pts[I_Mes]} - \text{Lat_Min}}{\text{Lat_Step}}\right)$ (3)

  Calculation of the mean latitude of the grid cell :

  - Lat_Cell = Lat_Min + Lat_Step/2 + J_Lat∗Lat_Step (4)

  Calculation of the longitude step at the latitude of the grid cell:

  - $\text{Lon_Step} = \frac{\text{Dist_Lon_Step} * 360}{2\pi R * \cos(\text{Lat_Cell})}$ (5)

    with R = Earth radius at a mean 70° latitude : R = 6370000 m

  Calculation of the longitude index of the grid cell :

  - $\text{I_Lon} = \text{INT}\left(\frac{\text{Lon_Pts[I_Mes]} - \text{Lon_Min}}{\text{Lon_Step}}\right)$ (6)

  Calculation of the grid cell index :

  - $\text{I_Cell} = \text{I_Lon} + \sum_{L=0}^{\text{J_Lat}-1} \text{Tab_Nb_Lon}(L)$ (7)

  Checking if the measurement point is to be included in the averaging process, according to its value:

  - Val_Flag = 0 (8)
  - If Threshold_Min ≠ Threshold_Max (i.e., if selection on the parameter values is required):

    ∗ If ((Val_Pts[I_Mes] < Threshold_Min) OR (Val_Pts[I_Mes] > Threshold_Max)) then:

    Val_Flag = 1 (9)
  - If Val_Flag = 0, then :

    Distribution of the parameter value in its cell :

    Val_Pts_Per_Cell[I_Cell][Nb_Pts_Per_Cell[I_Cell]] = Val_Pts[I_Mes] (10)

    Nb_Pts_Per_Cell[I_Cell] = Nb_Pts_Per_Cell[I_Cell] + 1 (11)

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- For I_Cell = 0 to Nb_Cells-1
  - Grid_Nb_Points[I_Cell] = Nb_Pts_Per_Cell[I_Cell] (12)
  - The minimum value, maximum value, mean value and standard deviation of the parameter is computed in each cell, using the mechanism “GEN_MEC_COM_01 - Arithmetic averaging”

    The input parameters of which are :

    ∗ N = Nb_Pts_Per_Cell[I_Cell]

    ∗ Val[0:N-1] = Val_Pts_Per_Cell[I_Cell][0:Nb_Pts_Per_Cell[I_Cell]]

    And the output parameters of which are :

    ∗ Mean = Grid_Mean[I_Cell]

    ∗ Std = Grid_Std[I_Cell]

    ∗ Min = Grid_Mini[I_Cell]

    ∗ Max = Grid_Maxi[I_Cell]

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_INT_01 - Linear weighting

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J. STUM CLS	
Checked by:	For the JASON-1 SWT C.K. SHUM For the JASON-1 Project P. VINCENT CNES S. DESAI JPL For the F-PAC / ENVISAT Project J. BENVENISTE ESA P. VINCENT CNES	
Approved by:	P. VINCENT CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To compute the indexes of the two elements of a one-dimensional table and their corresponding weights to be used in a linear interpolation.

ALGORITHM SPECIFICATION

Input data

- Minimum value of the table : Xmin
- Maximum value of the table : Xmax
- Table step : Xstep
- Abscissa where linear interpolation between the first and the second table values will occur : X

Output data

- Index of the first table value : Index_1
- Index of the second table value : Index_2
- Linear weight corresponding to the first value : W1 (/)
- Linear weight corresponding to the second value : W2 (/)
- Execution status

Processing

- If $X \leq Xmin$, then:
  - Index_1 = 0 (1)
  - W1 = 1 (2)
  - Index_2 = Index_1 (3)
  - W2 = 0 (4)
- If $X \geq Xmax$, then:
  - Index_2 = INT[(Xmax – Xmin)/Xstep] (5)
  - W2 = 1 (6)
  - Index_1 = Index_2 (7)
  - W1 = 0 (8)

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- If Xmin < X < Xmax, then:
  - Index_1 = INT[(X – Xmin)/Xstep] (9)
  - Index_2 = Index_1 + 1 (10)
  - $W1 = \frac{Xmin + Index_2 * Xstep - X}{Xstep}$ (11)
  - W2 = 1 – W1 (12)

COMMENTS

None

REFERENCES

None

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_INT_02 - Linear interpolation

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J. STUM CLS	
Checked by:	For the JASON-1 SWT C.K. SHUM For the JASON-1 Project P. VINCENT CNES S. DESAI JPL For the F-PAC / ENVISAT Project J. BENVENISTE ESA P. VINCENT CNES	
Approved by:	P. VINCENT CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To perform a linear interpolation between two values

ALGORITHM SPECIFICATION

Input data

- First value : X1
- Second value : X2
- Weight to be applied to the first value : W1 ([0,1]) (/)
- Weight to be applied to the second value : W2 ([0,1]) (/)

Output data

- Value resulting from linear interpolation between the first and the second values : PARAM (same unit as X)
- Execution status

Processing

$$PARAM = W1 * X1 + W2 * X2 \quad (1)$$

COMMENTS

None

REFERENCES

None

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_INT_03 - Bilinear interpolation

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	P. SICARD CLS	
Checked by:	For the JASON-1 SWT C.K. SHUM For the JASON-1 Project P. VINCENT CNES S. DESAI JPL For the F-PAC / ENVISAT Project J. BENVENISTE ESA P. VINCENT CNES	
Approved by:	P. VINCENT CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To perform bilinear interpolation.

ALGORITHM SPECIFICATION

Input data

- Values of the four corners of the cell:
  - Value of the lower left corner : Val_LL
  - Value of the lower right corner : Val_LR
  - Value of the upper left corner : Val_UL
  - Value of the upper right corner : Val_UR
- Weights of the four corners of the cell:
  - Weight of the lower left corner : W_LL ([0,1]) (/)
  - Weight of the lower right corner : W_LR ([0,1]) (/)
  - Weight of the upper left corner : W_UL ([0,1]) (/)
  - Weight of the upper right corner: : W_UR([0,1]) (/)
- Default value of a corner : NIL [(1)]

Output data

- Tabulated parameter interpolated : Interp_Value (same unit as Val)
- Number of valid cell points used by the interpolation : Nb_Pts ([0,4]) (/)
- Execution status

Processing

- If NIL is set to NaN or if NIL > 0, then:
  - Computing the real weight of each corner with elimination of the cell points having default value:
    - Nb_Pts=0
    - If (VAL_LL ≠ NIL) then Nb_Pts = Nb_Pts+1, else W_LL = 0 and VAL_LL = 0 (1)
    - If (VAL_LR ≠ NIL) then Nb_Pts = Nb_Pts+1, else W_LR = 0 and VAL_LR = 0 (2)
    - If (VAL_UL ≠ NIL) then Nb_Pts = Nb_Pts+1, else W_UL = 0 and VAL_UL = 0 (3)
    - If (VAL_UR ≠ NIL) then Nb_Pts = Nb_Pts+1, else W_UR = 0 and VAL_UR = 0 (4)

---

(1) 0 if there is no default value, > 0 (for integers) or NaN (for reals) if the default value exists

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- Else (NIL = 0):
  - Nb_Pts = 4
- Processing the bilinear interpolation if at least one corner of the cell is valid. Otherwise the tabulated parameter is not computed:

  If (Nb_Pts $\neq$ 0), then:

$$\text{Interp_Value} = \frac{\text{Val_LL} * \text{W_LL} + \text{Val_LR} * \text{W_LR} + \text{Val_UL} * \text{W_UL} + \text{Val_UR} * \text{W_UR}}{\text{W_LL} + \text{W_LR} + \text{W_UL} + \text{W_UR}} \quad (6)$$

COMMENTS

None

REFERENCES

None

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_INT_04 - Bivariate interpolation from six points

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J.P. DUMONT CLS	
Checked by:	For the JASON-1 SWT C.K. SHUM For the JASON-1 Project P. VINCENT CNES S. DESAI JPL For the F-PAC / ENVISAT Project J. BENVENISTE ESA P. VINCENT CNES	
Approved by:	P. VINCENT CNES	

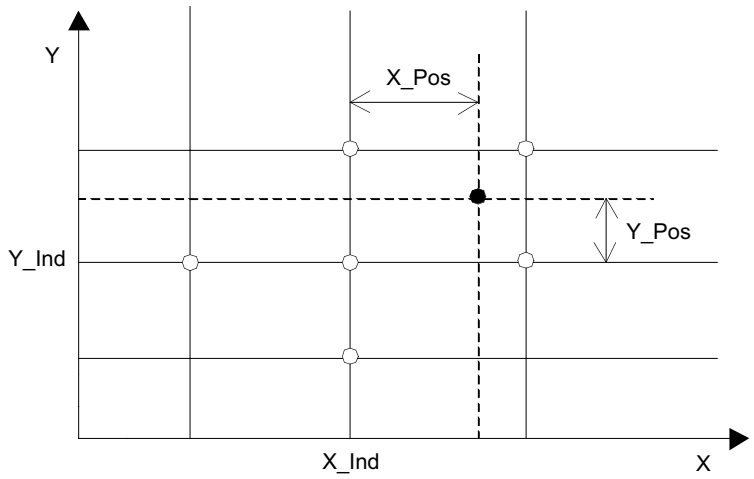
Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To perform a bivariate interpolation from six points.



ALGORITHM SPECIFICATION

Input data

- Relative position in the cell (to X_Ind) : X_Pos (/) (∈ [0, 1[)
- Relative position in the cell (to Y_Ind) : Y_Pos (/) (∈ [0, 1[)
- Values of the table for the six points:
  - Point (x-,y) : Tab(X_Ind-1, Y_Ind)
  - Point (x,y-) : Tab(X_Ind, Y_Ind-1)
  - Point (x,y) : Tab(X_Ind, Y_Ind)
  - Point (x,y+) : Tab(X_Ind, Y_Ind+1)
  - Point (x+,y) : Tab(X_Ind+1, Y_Ind)
  - Point (x+,y+) : Tab(X_Ind+1, Y_Ind+1)

Output data

- Interpolated value of the table : Tab_Interp (same unit as Tab)
- Execution status

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Processing

$$
\begin{aligned}
Tab_Interp = & \frac{Y_Pos * (Y_Pos - 1)}{2} * Tab(X_Ind, Y_Ind - 1) + \frac{X_Pos * (X_Pos - 1)}{2} * Tab(X_Ind - 1, Y_Ind) \\
& + \left(1 + X_Pos * Y_Pos - X_Pos^2 - Y_Pos^2\right) * Tab(X_Ind, Y_Ind) \\
& + \frac{X_Pos * (X_Pos - 2 * Y_Pos + 1)}{2} * Tab(X_Ind + 1, Y_Ind) \\
& + \frac{Y_Pos * (Y_Pos - 2 * X_Pos + 1)}{2} * Tab(X_Ind, Y_Ind + 1) \\
& + X_Pos * Y_Pos * Tab(X_Ind + 1, Y_Ind + 1)
\end{aligned}
\tag{1}
$$

COMMENTS

None

REFERENCES

- Abramowitz M. and Stegun I.A., Hanbook of Mathematical Functions, Dover Publication Inc. N.Y., 1965 (p. 882, section 25.2.67)

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_INT_05 - Interpolation of an array at a new sampling frequency

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	P. THIBAUT	CLS	
Checked by:	N. PICOT	CNES	
Approved by:	P. VINCENT	CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To interpolate an array of values at a new sampling frequency.

ALGORITHM SPECIFICATION

Input data

- Number of samples of the input array : Num_Val_In_X (/)
- Interval (in frequency) between two samples of the input array : T_In_X (Hz)
- Index associated to the zero frequency of the input array : F0_In_X (/)
- Samples of the input array : Val_In_X [0:Num_Val_In_X-1] (1)
- Interval (in frequency) between two samples of the output array : T_Out_X (Hz)

Output data

- Number of samples of the output array : Num_Val_Out_X (/)
- Samples of the output array : Val_Out_X [0:Num_Val_Out_X-1] (1)
- Index associated to the zero frequency of the output array : F0_Out_X (/)
- Execution status

Processing

This algorithm is aimed at interpolating the input array at a new sampling frequency that is greater than the sampling frequency of the array.

- $C = \mathrm{NINT}\left[\frac{T_In_X}{T_Out_X}\right]$ (1)
- $Num_Val_Out_X = C * Num_Val_In_X$ (2)
- $F0_Out_X = C * F0_In_X$ (3)
- Frequencies of the input array:

  $Freq_In_j = (j - F0_In_X) * T_In_X$ for j from 0 to Num_Val_In_X-1 (4)

- Frequencies of the output array:

  $Freq_Out_j = (j - F0_Out_X) * T_Out_X$ (5)

(1) FFT power unit if the input array is the PTR and no unit if the input vector if the filter

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- Frequency corresponding to the first point in the input array:

$$FreqFirst = (-F0_In_X) * T_In_X \quad (6)$$

- Frequency corresponding to the last point in the input array:

$$FreqLast = (Num_Val_In_X - 1 - F0_In_X) * T_In_X \quad (7)$$

- For each j (j from 0 to Num_Val_Out_X-1) of the output array, compute the interpolated value, using mechanisms "GEN_MEC_INT_02 - Linear interpolation" and "GEN_MEC_INT_01 - Linear weighting".
  - The inputs for "GEN_MEC_INT_01 - Linear weighting" are:
    - Abscissa where linear interpolation between the first and the second values will occur : Freq_Outj (Hz)
    - Abscissa corresponding to the first value of the input table : FreqFirst (Hz)
    - Abscissa corresponding to the second value of the input table : FreqLast (Hz)
    - Table step : T_In_X (Hz)

    The outputs are:
    - Index corresponding to the first value (of the input table) : J_Before (/)
    - Index corresponding to the second value (of the input table) : J_After (/)
    - Linear weight corresponding to the first value : WBefore (/)
    - Linear weight corresponding to the second value : WAfter (/)
  - The inputs for "GEN_MEC_INT_02 - Linear interpolation" are:
    - First value : Val_In_X[J_Before]
    - Second value : Val_In_X [J_After]
    - Weight to be applied to the first value (output of "Linear weighting") : WBefore (/)
    - Weight to be applied to the second value (output of "Linear weighting") : WAfter (/)

    The output is:
    - Value resulting from linear interpolation between the first and the second values: Val_Out_X [j] (same unit as X)
- On the right side of the (Num_Val_In_X-1)th sample of the input array, C-1 points have to be extrapolated. Because these samples will not be taken into account in the following of the processing, extrapolated values are set to the value of the last point of the input array.

COMMENTS

None

REFERENCES

None

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_INT_06 - Everett interpolation

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J.P. DUMONT CLS	
Checked by:	For the JASON-1 SWT R. SCHAROO F. LEMOINE For the JASON-1 Project P. VINCENT CNES S. DESAI JPL For the F-PAC / ENVISAT Project J. BENVENISTE ESA P. VINCENT CNES	
Approved by:	P. VINCENT CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To interpolate an array of equidistant scalar values using Everett formula (with an order smaller or equal to 14).

ALGORITHM SPECIFICATION

Input data

- Number of values in the array be interpolated [1] : N (/)
- Array of scalar values to be interpolated : Y[0:N-1]
- Abscissa of the (N/2+1)th element of the array : X
- Sampling step of the array : X_Step (same unit as X)
- Abscissa for the interpolation : X_Ref (same unit as X)

Output data

- Interpolated value of the array : Y_Int (same unit as Y)
- Execution status

Processing

$$Y_Int = \sum_{i=0}^{\frac{N}{2}-1} \left[U(i) * R(i) + V(i) * S(i)\right] \quad (1)$$

with the various parameters are computed as follows:

1) $R(0) = \dfrac{X - X_Ref}{X_Step}$ and $S(0) = 1 - R(0)$ (2)

2) $Z(8 - \dfrac{N}{2} + i) = Y(i)$, for $i \in [0, N-1]$ (3)

3) If $\dfrac{N}{2} = 1$ then go to step 4. Else, parameters R(i) and S(i) for $i \in [1, \dfrac{N}{2} -1]$ are computed using the following formula (with W = R or S):

[1] Order of the Everett formula = Number of values for the interpolation - 2

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

$$W(i) = W(i-1) * \frac{W(0)^2 - i^2}{4 * i^2 + 2 * i} \tag{4}$$

4) Parameters U(i) and V(i) are computed as follows for $i \in [0, \frac{N}{2} -1]$:

$$\begin{aligned} U(0) &= Z(7) \\ V(0) &= Z(8) \end{aligned} \tag{5}$$

$$\begin{aligned} U(1) &= [Z(8) - U(0)] + [Z(6) - U(0)] \\ V(1) &= [Z(9) - V(0)] + [Z(7) - V(0)] \end{aligned} \tag{6}$$

$$\begin{aligned} U(2) &= [Z(9) - U(0)] + [Z(5) - U(0)] - 4 * U(1) \\ V(2) &= [Z(10) - V(0)] + [Z(6) - V(0)] - 4 * V(1) \end{aligned} \tag{7}$$

$$\begin{aligned} U(3) &= [Z(10) - U(0)] + [Z(4) - U(0)] - 6 * U(2) - 9 * U(1) \\ V(3) &= [Z(11) - V(0)] + [Z(5) - V(0)] - 6 * V(2) - 9 * V(1) \end{aligned} \tag{8}$$

$$\begin{aligned} U(4) &= [Z(11) - U(0)] + [Z(3) - U(0)] - 8 * U(3) - 20 * U(2) - 16 * U(1) \\ V(4) &= [Z(12) - V(0)] + [Z(4) - V(0)] - 8 * V(3) - 20 * V(2) - 16 * V(1) \end{aligned} \tag{9}$$

$$\begin{aligned} U(5) &= [Z(12) - U(0)] + [Z(2) - U(0)] - 10 * U(4) - 35 * U(3) - 50 * U(2) - 25 * U(1) \\ V(5) &= [Z(13) - V(0)] + [Z(3) - V(0)] - 10 * V(4) - 35 * V(3) - 50 * V(2) - 25 * V(1) \end{aligned} \tag{10}$$

$$\begin{aligned} U(6) &= [Z(13) - U(0)] + [Z(1) - U(0)] - 12 * U(5) - 54 * U(4) - 112 * U(3) - 105 * U(2) - 36 * U(1) \\ V(6) &= [Z(14) - V(0)] + [Z(2) - V(0)] - 12 * V(5) - 54 * V(4) - 112 * V(3) - 105 * V(2) - 36 * V(1) \end{aligned} \tag{11}$$

$$\begin{aligned} U(7) &= [Z(14) - U(0)] + [Z(0) - U(0)] - 14 * U(6) - 77 * U(5) - 210 * U(4) - 294 * U(3) - 196 * U(2) - 49 * U(1) \\ V(7) &= [Z(15) - V(0)] + [Z(1) - V(0)] - 14 * V(6) - 77 * V(5) - 210 * V(4) - 294 * V(3) - 196 * V(2) - 49 * V(1) \end{aligned} \tag{12}$$

COMMENTS

None

REFERENCES

None

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_INT_07 - Spline interpolation

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J. STUM	CLS	
Checked by:	N. PICOT	CNES	
Approved by:	P. VINCENT	CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To compute a parameter Y at a given abscissa X, by spline interpolation over 2N points around the given abscissa (nominally N points before and N points after the given abscissa). This mechanism utilizes standard routines of the NAG library.

ALGORITHM SPECIFICATION

Input data

- Number of points for the spline calculation (even) : N_Sp (/)
- Abscissas of the points : X_In[0:N_Sp-1]
- Ordinates of the points : Y_In[0:N_Sp-1]
- Smoothing factor : S [(1)]
- Abscissa at which interpolation is required : X_Int

Output data

- Ordinate interpolated at the required abscissa : Y_Int
- Sum of squares of residuals : FP [(1)]
- Status of the routines of the NAG library : IFAIL
- Execution status

Processing

- Abscissas and ordinates are computed relative to the values at point N_Sp/2 – 1, as follows :
  - For I = 0 to N_Sp – 1 :

    X_Rel[I] = X_In[I] – X_In[N_Sp/2 – 1] (1)

    Y_Rel[I] = Y_In[I] – Y_In[N_Sp/2 – 1] (2)

    Weight[I] = 1 (weights required for cubic spline interpolation) (3)
  - X_Int_Rel = X_Int – X_In[N_Sp/2 – 1] (4)

[(1)] Same unit as $(Y_In)^2$

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- The cubic spline approximation is computed on the set of N_Sp points, using the routine E02BEF of the NAG library,

  the parameters of which are:

  - START (input) = 'C'
  - M (input) = N_Sp
  - X (input) = X_Rel
  - Y (input) = Y_Rel
  - W (input) = Weight
  - S (input)
  - NEST (input) = M+4
  - N (output) : The total number of knots of the computed spline
  - LAMDA (output) : The positions of the knots (array of NEST points)
  - C (output) : The coefficients of the computed spline (array of NEST points)
  - FP (output) : The sum of squares of residuals
  - WRK (input) : Unused (array of LWRK points)
  - LWRK (input) = 4M + 16NEST + 41
  - IWRK (input) : Unused (array of NEST points)
  - IFAIL (input/output) = -1 on input
- If IFAIL $\neq$ 0, then:
  - exit
- The value of the cubic spline approximation at the required abscissa Y_Rel_Int is computed, using the routine E02BBF of the NAG library,

  the parameters of which are:

  - NCAP7 (input) = N
  - LAMDA (input) = LAMDA
  - C (input) = C
  - X (input) = X_Int_Rel
  - S (output) = Y_Rel_Int
  - IFAIL (input/output) = -1 on input
- The interpolated ordinate is then derived from the interpolated relative ordinate :
  - Y_Int = Y_Rel_Int + Y_In[N_Sp/2 – 1] (5)

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_INT_08 - One-dimensional polynomial interpolation

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	S. LABROUE	CLS	
Checked by:	N. PICOT	CNES	
Approved by:	P. VINCENT	CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	Creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To calculate the interpolation coefficients required for a fixed degree polynomial interpolation along one dimension. The interpolation is performed on a regular grid.

ALGORITHM SPECIFICATION

Input data

- Degree of the interpolation : Deg (/)
- Position of the interpolated point : X (/)
- Grid of interpolation
  - Minimum value of the grid : X_min (/)
  - Grid step : Step (/)
  - Number of grid values : N (/)

Output data

- Validity flag for the processing : Flag_int [(1)]
- Degree of the interpolation performed : Deg_int (/)
- Interpolation coefficients : Coeff[0:Deg_int -1] (/)
- Indexes of the grid points used in the interpolation : Ind[0:Deg_int-1] (/)

Processing

The following formula (the Lagrange formula) is used to perform one-dimensional polynomial interpolation of degree M (i.e. using M points) :

$$Y(X) = \sum_{k=0}^{M-1} Y(X_k) \prod_{\substack{i=0 \\ i \neq k}}^{M-1} \frac{X - X_i}{X_k - X_i} \qquad (1)$$

where: $Y(X)$ is the interpolated value at point X

$Y(X_k)$ are the values at grid points $X_k, k = 0, M-1$

(1) 2 states: "valid" or "invalid"

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Using a regular grid, this formula is written under the following form :

$$Y(X) = \sum_{k=0}^{M-1} Y(X_k) \prod_{i=0, i\neq k}^{M-1} \frac{X - [X_min + i * Step]}{(k - i) * Step} = \sum_{k=0}^{M-1} Y(X_k) * Coeff(k) \quad (2)$$

The following processing is performed to compute the interpolation coefficients Coeff(k), k = 1, M and the indexes k of the grid points used in the interpolation :

- To check the validity of the degree of interpolation :

  Deg_int = Deg (3)

  – If Deg > N then Deg_int = N (4)

  – If Deg is odd then Deg_int = Deg-1 (5)

- To compute the index j of the grid point preceding X

  j = INT[(X – X_min) / Step] (6)

- Boundary case : to perform linear interpolation at the boundaries of the grid

  If j > (N - 1 - Deg_int /2) or j < Deg_int /2 –1 then (7)

  – Deg_int = 2 (8)

  – If j >= N-1 then j = N-2 (9)

  – If j < 0 then j = 0 (10)

  – Ind(0) = j and Ind(1) = j+1 (11)

- General case : to calculate the indexes of the grid points used in the interpolation Ind(i), i = 0, Deg_int -1

  Else

  i = 0, Deg_int-1

  Ind(i) = j – Deg_int /2 + i + 1 (12)

- To compute the interpolation coefficients Coeff(i), i = 0, Deg_int -1 :

  – r = Ind(0) (13)

  – i = 0, Deg_int -1

  Coeff (i) = 1 (14)

  $$Coeff\ (i) = \prod_{j=r, j\neq r+i}^{r+Deg_int-1} \frac{X - [X_min + j * Step]}{(r + i - j) * Step} \quad (15)$$

- Flag_int is set to “valid”

COMMENTS

The interpolated value is not computed in this function.

The accuracy is linked to the degree of interpolation performed. It is better to choose low degree interpolation that is to say 2 or 4.

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

REFERENCES

"Numerical recipes in C, the Art of Scientific computing, Second Edition", William H. Press, Saul A. Teukolsky, William T. Vetterling, Brian P. Flannery, 1992, p.108

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_INT_09 - Computation of the coefficients of a polynomial interpolation on a two dimension grid

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	S. LABROUE	CLS	
Checked by:	N. PICOT	CNES	
Approved by:	P. VINCENT	CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To calculate the interpolation coefficients required for a fixed degree polynomial interpolation on a two dimension grid. The interpolation is performed on a regular grid.

ALGORITHM SPECIFICATION

Input data

- Degree of interpolation along dimension i : Deg_i (/)
- Degree of interpolation along dimension j : Deg_j (/)
- Abscissa of the interpolated point: : Xi (/)
- Ordinate of the interpolated point: : Xj (/)
- Offset on the limits of the grid in dimension j : Offset_j (same unit as Xj)
- Grid of interpolation:
  - For dimension i
    - Minimum value of the grid : Xi_min (/)
    - Grid step : Path_i (/)
    - Number of grid values : Nb_i (/)
  - For dimension j
    - Minimum value of the grid : Xj_min (/)
    - Grid step : Path_j (/)
    - Number of grid values : Nb_j (/)

Output data

- Validity flag for the processing : Flag_int [(1)]
- Interpolation coefficients : Coeff[0:(Nb_i*Nb_j) -1] (/)
- Index of the first grid point used in the interpolation : Ind_min (/)
- Index of the last grid point used in the interpolation : Ind_max (/)

Processing

To interpolate a point on a two dimension grid, the basic idea is to break up the problem into a succession of one-dimensional interpolations.

---

(1) 2 states: "valid" or "invalid"

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

To perform an interpolation of degree M in the direction j and of degree N in the direction i, an M*N sub-block of points of the grid that contains the point to interpolate, is used. The processing of interpolation is reduced to realize M one-dimensional interpolations in direction i and a last interpolation is then performed in direction j to get the interpolated value.

The interpolation of the point $X(Xj, Xi)$ gives the following equations :

- Polynomial interpolation of degree N (using N points) in direction i

$$X(j, Xi) = \sum_{i=0}^{N-1} a_i \, X(j,i) \tag{1}$$

where

$j = 0, M-1$ are the indexes of the grid in direction j

$i = 0, N-1$ are the indexes of the grid in direction i

$X(j,i)$ is the value at grid point (j,i)

- Polynomial interpolation of degree M (using M points) in direction j

$$X(Xj, Xi) = \sum_{j=0}^{M-1} b_j \, X(j, Xi) \tag{2}$$

Finally, the interpolated value is expressed by:

$$X(Xj, Xi) = \sum_{j=0}^{M-1} \sum_{i=0}^{N-1} a_i \, b_j \, X(j,i) \tag{3}$$

If the MN points of the grid used in the interpolation have global indexes k such that $k \in [k_1, k_{MN}]$, formula (3) can be written under the following form:

$$X(Xj, Xi) = \sum_{k=k_1}^{k_{MN}} \alpha_k X_k \tag{4}$$

with $\alpha_k = a_i \, b_j$ (5)

$k = i * M + j$ (6)

Formula (6) is valid for a numeration of the grid points increasing in direction j.

The processing to compute the interpolation coefficients and their indexes in the grid is specified hereafter :

To test if the input point (Xi, Xj) belong to the grid which limits are increased by an offset on dimension j :

- If (Xi < Xi_min) or (Xi > Xi_min + (Nb_i –1)*Step_i) or

  (Xj ≤ Xj_min - Offset_j) or (Xj ≥ Xj_min + (Nb_j –1)*Step_j + Offset_j) then

  the flag Flag_int is set to “invalid” and the outputs are set to default values.

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Else

– Interpolation in direction i :

Using mechanism “GEN_MEC_INT_08 - One-dimensional polynomial interpolation”, with inputs :

Deg_i, Xi, Xi_min, Step_i, Nb_i, the following outputs are performed :

Flag of interpolation validity : Flag_interp_i

Number of grid points used in the interpolation : Ni

Interpolation coefficients used in the interpolation : Coeff_i [0:Ni-1]

Indices of grid points used in the interpolation : Ind_i[0:Ni-1]

– Interpolation in direction j :

Using mechanism “GEN_MEC_INT_08 - One-dimensional polynomial interpolation”, with inputs :

Deg_j, Xj, Xj_min, Step_j, Nb_j, the following outputs are performed :

Flag of interpolation validity : Flag_interp_j

Number of grid points used in the interpolation : Nj

Interpolation coefficients used in the interpolation : Coeff_j [0:Nj-1]

Indices of grid points used in the interpolation : Ind_j[0:Nj-1]

– If Flag_interp_i and Flag_interp_j are “valid” then

∗ To calculate the interpolation coefficients for both dimensions i and j

For m = 0, Ni-1

For n = 0, Nj-1

◊ To compute the global indexes associated to the coefficients from the indexes of the interpolation coefficients in direction i and j, with the following formula (valid for a numeration of the grid increasing in direction j) :

$$k = (Ind_i(m) - 1) * Nb_j + Ind_j(n) \quad (7)$$

◊ The interpolation coefficients (Coeff{k}) are computed by :

$$Coeff(k) = Coeff_i(m) * Coeff_j(n) \quad (8)$$

∗ To determine the first and last indexes of the grid points used in the interpolation :

$$Ind_min = (Ind_i(0) - 1) * Nb_j + Ind_j(0) \quad (9)$$

$$Ind_max = (Ind_i(Ni-1) - 1) * Nb_j + Ind_j(Nj-1) \quad (10)$$

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- Flag of the function (Flag_int) is set to “valid”

– Else Flag_int is set to “invalid”

COMMENTS

None

REFERENCES

“Numerical recipes in C, the Art of Scientific computing, Second Edition”, William H. Press, Saul A. Teukolsky, William T. Vetterling, Brian P. Flannery, 1992, p.123

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_INT_10 - Interpolation of altimeter measurements at a given point along-track

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J. STUM	CLS	
Checked by:	N. PICOT	CNES	
Approved by:	P. VINCENT	CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To interpolate altimeter measurements at a given point along-track. Each altimeter measurement contains different fields. Some fields are linearly interpolated, other fields are spline interpolated (the orbit height and the altimeter ranges), other fields are set to the value of the closest altimeter point (the flags and the numbers of elementary measurements).

ALGORITHM SPECIFICATION

Input data

- The time tag of the given point along-track : Time_Int (s)
- The index of the altimeter measurement which time is the closest and greater than the time tag of the point along-track : Index_Higher (/)
- The minimum value of index of altimeter measurement to consider for the search of N_Sp altimeter measurements surrounding the point along-track : Index_Min (/)
- The maximum value of index of altimeter measurement to consider for the search of N_Sp altimeter measurements surrounding the point along-track : Index_Max (/)
- The number of points used for spline interpolation : N_Sp (/)
- The time tags of the altimeter measurements :Time[Index_Min:Index_Max] (s)
- The time interval between two consecutive measurements : Delta_Time (s)
- The weight for linear interpolation, to be applied to measurement number Index_Higher : W (/)
- The number of fields to be interpolated (linearly or by spline) : N_Int (/)
- The number of fields to be set to the closest value : N_Clos (/)
- The number of fields to be interpolated by spline : N_Spl (/)
- The field numbers to be interpolated : FN_Int[0:N_Int-1] (/)
- The field numbers to set set to the closest value : FN_Clos[0:N_Clos-1] (/)
- The field numbers to be spline interpolated : FN_Spl[0:N_Spl-1]
- The smoothing factors of the fields to be spline interpolated : SF[0:N_Spl-1] (m)
- The altimeter measurements : Alt_Meas[Index_Min:Index_Max][0:N_Int+N_Clos-1]

Output data

- The interpolated altimeter measurements : Alt_Meas_Int[0:N_Int+N_Clos-1]
- The interpolation flag (0 = spline interpolation with N_Sp/2 measurements before and N/Sp measurements after, 1 = spline interpolation with N_Sp measurements but with less than N_Sp/2 measurements on one side, 2 = linear interpolation) : Flag_Spline (/)
- The RMS of fit of the N_Spl fields for which spline interpolation was performed : RMS_Of_Fit[0:N_Spl-1] (m)
- Execution status

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Processing

Linear interpolation of all the parameters, excepted the flags and the numbers of elementary measurements. This linear interpolation includes the parameters to be spline interpolated :

- For J = 0 to N_Int-1
  - If Alt_Meas[Index_Higher-1][FN_Int[J]-1] is set to its default value or if Alt_Meas[Index_Higher][FN_Int[J]-1] is set to its default value, then :
    - Alt_Meas_Int[FN_Int[J]-1] is set to its default value
  - Else, computing by linear interpolation the parameter at the point along-track, using the mechanism "GEN_MEC_INT_02 - Linear interpolation",

    the input parameters of which are :
    - X1 = Alt_Meas[Index_Higher- 1][FN_Int[J]-1]
    - X2 = Alt_Meas[Index_Higher][FN_Int[J]-1]
    - W1 = 1 - W
    - W2 = W

    and the output parameter of which are :
    - Param_Int = Alt_Meas_Int[FN_Int[J]-1]
    - The execution status

Setting the flags and the numbers of elementary measurements to the values of the closest altimeter point :

- For K = 0 to N_Clos-1
  - If W > 0.5, then :
    - Alt_Meas_Int[FN_Clos[K]-1] = Alt_Meas[Index_Higher][FN_Clos[K]-1] (1)
  - Else :
    - Alt_Meas_Int[FN_Clos[K]-1] = Alt_Meas[Index_Higher - 1][FN_Clos[K]-1] (2)

Search of the N actual points around the point along-track for spline interpolation :

- Compute the beginning and end indexes of the actual altimeter measurements to consider for spline interpolation, using the mechanism “GEN_MEC_SEL_01 – Selection of N points for spline interpolation”,

  The input parameters of which are :
  - Index_Higher
  - Index_Min
  - Index_Max
  - N = N_Sp
  - Time[Index_Min:Index_Max]
  - Delta_Time

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

And the output parameters of which are :

- Index_First = Index_Beg
- Index_Last = Index_End
- Flag_Avail
- The execution status

If the N altimeter measurements around the point along-track are available (Flag_Avail < 2), spline interpolate the orbit height and the altimeter ranges :

- Flag_Spline = 2 ("linear interpolation performed")
- For L = 0 to N_Spl – 1
  - RMS_Of_Fit[L] is set to its default value
- If Flag_Avail < 2, then :
  - For L = 0 to N_Spl –1
    - Spline interpolate the parameter at the point along-track, using the mechanism "GEN_MEC_INT_07 – Spline interpolation",

      The input parameters of which are :

      ◊ N_Sp

      ◊ X_In[0:N_Sp – 1] = Time[Index_Beg:Index_End]

      ◊ Y_In[0:N_Sp – 1] = Alt_Meas[Index_Beg:Index_End] [FN_Spl[L]-1]

      ◊ S = SF[L]

      ◊ X_Int = Time_Int

      And the output parameters of which are :

      ◊ Y_Int

      ◊ FP

      ◊ IFAIL
    - If IFAIL = 0, then :

      ◊ Alt_Meas_Int[FN_Spl[L]-1] = Y_Int

      ◊ Flag_Spline = Flag_Avail

      Computing the RMS of fit from the sum of squares of differences (spline – Y_In) FP :

      ◊ $\text{RMS_Of_Fit[L]} = \sqrt{\frac{FP}{N_Sp}}$ (3)

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_INT_11 - Interpolation of 20-Hz altimeter measurements at a given point along-track

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J. STUM	CLS	
Checked by:	N. PICOT	CNES	
Approved by:	P. VINCENT	CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To interpolate 20-Hz altimeter measurements at a given point along-track. Each altimeter measurement contains different fields. Some fields are linearly interpolated (the geophysical corrections), other fields are interpolated by polynomial regression (the orbit height and all the retracking estimates).

ALGORITHM SPECIFICATION

Input data

- The time tag of the given point along-track : Time_Int (s)
- The index of the altimeter measurement which time is the closest and greater than the time tag of the point along-track : Index_Higher (/)
- The minimum value of index of altimeter measurement to consider for the polynomial interpolation of altimeter measurements surrounding the point along-track : Index_Min (/)
- The maximum value of index of altimeter measurement to consider for the polynomial interpolation of altimeter measurements surrounding the point along-track : Index_Max (/)
- The time tags of the altimeter measurements :Time[Index_Min:Index_Max] (s)
- The time interval between two consecutive measurements : Delta_Time (s)
- The weight for linear interpolation, to be applied to measurement number Index_Higher : W (/)
- The number of fields to be linearly interpolated : N_Lin (/)
- The number of fields to be interpolated by polynomial regression : N_Poly (/)
- The field numbers to be linearly interpolated : FN_Lin[0:N_Lin-1] (/)
- The field numbers to be interpolated by polynomial regression : FN_Poly[0:N_Poly-1]
- The altimeter measurements : Alt_Meas[Index_Min:Index_Max][0:N_Lin+N_Poly-1]
- The bandwidth of the Ku chirp band (0 : 320 MHz, 1 : 80 MHz, 2 : 20 MHz) : Bandwidth[Index_Min:Index_Max] (/)

Output data

- The interpolated altimeter measurements : Alt_Meas_Int[0:N_Lin+N_Poly-1]
- The slopes of the polynomial fits : Slope[0:N_Poly – 1]
- The curvatures of the polynomial fits : Curvature[0:N_Poly – 1]
- The bandwidth quality flag : Bandwidth_Flag (/)[1]
- Execution status

[1] 7 states : = 0 : all measurements 320 MHz. = 1 : 320 MHz + 80 MHz. = 2 : 320 MHz + 20 MHz. = 3 : 320 MHz + 80 MHz + 20 MHz. = 4 : All measurements 80 MHz. = 5 : 80 MHz + 20 MHz. = 6 : All measurements 20 MHz

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Processing

Linear interpolation is performed for some of the parameters (the geophysical corrections) :

- For J = 0 to N_Lin-1
  - If Alt_Meas[Index_Higher-1][FN_Lin[J]-1] is set to its default value or if Alt_Meas[Index_Higher][FN_Lin[J]-1] is set to its default value, then :
    - Alt_Meas_Int[FN_Lin[J]-1] is set to its default value (1)
  - Else, computing by linear interpolation the parameter at the point along-track, using the mechanism "GEN_MEC_INT_02 - Linear interpolation",

    the input parameters of which are :
    - X1 = Alt_Meas[Index_Higher- 1][FN_Lin[J]-1]
    - X2 = Alt_Meas[Index_Higher][FN_Lin[J]-1]
    - W1 = 1 - W
    - W2 = W

    and the output parameter of which are :
    - Param_Int = Alt_Meas_Int[FN_Lin[J]-1]
    - The execution status

Polynomial interpolation is performed for the other parameters (the orbit height and the retracking estimates):

- For J = 0 to N_Poly-1
  - Index_Sel = 0 (2)
  - For Index = Index_Min to Index_Max
    - If ABS(Time[Index] – Time[Index_Higher]) < Delta_Time∗ABS[Index – Index_Higher], then :
      - If Alt_Meas[Index][FN_Poly[J]-1] is not set to its default value, then :

        ⇒ Alt_Meas_Sel[Index_Sel] = Alt_Meas[Index][FN_Poly[J]-1] (3)

        ⇒ Time_Sel[Index_Sel] = Time[Index] (4)

        ⇒ Index_Sel = Index_Sel + 1 (5)
  - If Index_Sel > 2, then :
    - A polynomial least squares fit is performed, using the mechanism “GEN_MEC_COM_08 – Least-squares fit by polynomial or periodic functions”,

      the input parameters of which are :
      - NP = Index_Sel
      - X[0:NP-1] = Time_Sel[0:Index_Sel – 1]
      - Y[0:NP-1] = Alt_Meas_Sel[0:Index_Sel – 1]
      - Var[0:NP-1] = 1

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- ◊ Type_Fit = 'pol'
- ◊ Degree = 2
- ◊ Freq[0:Degree-1] = 0

and the output parameters of which are :

- ◊ Y_Adj[0:NP-1] = Alt_Meas_Adj[0:Index_Sel – 1]
- ◊ B[0:2∗Degree]

∗ The following parameters are computed at crossover time tag :

- ◊ Alt_Meas_Int[FN_Poly[J]-1] = B[0] + B[1]∗Time_Int + B[2]∗(Time_Int)^2 (6)
- ◊ Slope[J] = B[1] (7)
- ◊ Curvature[J] = B[2] (8)

– Else

∗ The following parameters are set to default value :

- ◊ Alt_Meas_Int[FN_Poly[J]-1]
- ◊ Slope[J]
- ◊ Curvature[J]

• The bandwidth quality flag is computed :

– Presence_320 = 0 (9)

– Presence_80 = 0 (10)

– Presence_20 = 0 (11)

– For Index = Index_Min to Index_Max

∗ If ABS(Time[Index] – Time[Index_Higher]) < Delta_Time∗ABS[Index – Index_Higher], then :

- ◊ If Bandwidth[Index] = 0, then

  ⇒ Presence_320 = 1 (12)
- ◊ If Bandwidth[Index] = 1, then

  ⇒ Presence_80 = 1 (13)
- ◊ If Bandwidth[Index] = 2, then

  ⇒ Presence_20 = 1 (14)

– If Presence_320 = 1 and Presence_80 = 0 and Presence_20 = 0, then :

∗ Bandwidth_Flag = 0 (15)

– If Presence_320 = 1 and Presence_80 = 1 and Presence_20 = 0, then :

∗ Bandwidth_Flag = 1 (16)

– If Presence_320 = 1 and Presence_80 = 0 and Presence_20 = 1, then :

∗ Bandwidth_Flag = 2 (17)

– If Presence_320 = 1 and Presence_80 = 1 and Presence_20 = 1, then :

∗ Bandwidth_Flag = 3 (18)

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- If Presence_320 = 0 and Presence_80 = 1 and Presence_20 = 0, then :
  - Bandwidth_Flag = 4 (19)
- If Presence_320 = 0 and Presence_80 = 1 and Presence_20 = 1, then :
  - Bandwidth_Flag = 5 (20)
- If Presence_320 = 0 and Presence_80 = 0 and Presence_20 = 1, then :
  - Bandwidth_Flag = 6 (21)

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_INT_12 - Spline interpolation of grid values

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J. STUM	CLS	
Checked by:	N. PICOT	CNES	
Approved by:	P. VINCENT	CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To interpolate at an altimeter measurement (given by its longitude and its latitude) field values provided on a regular grid in longitude and latitude, using one-dimensional spline functions. For each grid map column within an interpolation window, an interpolating spline is derived as function of grid point latitude, and is evaluated at the altimeter measurement latitude. The series of spline values are then used to build an interpolating spline as function of grid point longitude, which is evaluated at the altimeter measurement longitude to provide the final interpolated field value.

ALGORITHM SPECIFICATION

Input data

- Location of the measurement:
  - Longitude : Lon (degree or minute, positive)
  - Latitude : Lat (degree or minute)
- Grid map:
  - Longitude of first grid point : Lon_First (degree or minute, positive)
  - Grid step in longitude : Step_Lon (degree or minute)
  - Number of points in longitude : Nb_Pts_Lon (/)
  - Latitude of first grid point : Lat_First (degree or minute)
  - Grid step in latitude : Step_Lat (degree or minute)
  - Number of points in latitude : Nb_Pts_Lat (/)
  - Grid values : Grid[0:Nb_Pts_Lon-1][0:Nb_Pts_Lat-1]
  - Default value in Model : Def_Value (/)
  - Cycling value for the X variable : Xcyc (≥ 0, degree or minute)
- Number of grid points in longitude (and latitude) used for spline calculation : N_Window (/)

Output data

- Interpolated field at altimeter measurement : Field_Int
- Execution status

Processing

- If Lon < Lon_First, then :
  - Lon = Lon + Xcyc
- If N_Window is even, then :
  - $Ileft = INT(\frac{Lon - Lon_First}{Step_Lon})$

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- Jbot = INT($\frac{\text{Lat - Lat_First}}{\text{Step_Lat}}$)
- Offset = 1

- Else
  - Ileft = NINT($\frac{\text{Lon - Lon_First}}{\text{Step_Lon}}$)
  - Jbot = NINT($\frac{\text{Lat - Lat_First}}{\text{Step_Lat}}$)
  - Offset = 0

The interpolation window is selected within the grid, as follows :

- For I_Lon = 0 to N_Window – 1
  - I_Window = Ileft - $\frac{\text{N_Window}}{2}$ + Offset + I_Lon
  - If I_Window < 0, then :
    - I_Window = I_Window + Nb_Pts_Lon
  - If I_Window > Nb_Pts_Lon-1, then
    - I_Window = I_Window – Nb_Pts_Lon
  - Lon_Grid[I_Lon] = Lon_First + I_Window∗Step_Lon
  - For J_Lat = 0 to N_Window – 1
    - J_Window = Jbot - $\frac{\text{N_Window}}{2}$ + Offset + J_Lat
    - If J_Window < 0 or J_Window > Nb_Pts_Lat – 1, then :
      - ◊ Field_Int is set to its default value
      - ◊ Exit
    - Lat_Grid[J_Lat] = Lat_First + J_Window∗Step_Lat
    - Grid_Window[J_Lat] = Grid[I_Window][J_Window]
    - If Def_Value is set to NaN or if Def_Value > 0, then :
      - ◊ If Grid_Window[J_Lat] is set to its default value, then :
        - ⇒ Field_Int is set to its default value
        - ⇒ Exit
  - Spline interpolate the grid values at the latitude of the altimeter measurement, using the mechanism "GEN_MEC_INT_07 – Spline interpolation",

    The input parameters of which are :
    - ◊ N_Sp = N_Window
    - ◊ X_In[0:N_Sp – 1] = Lat_Grid[0:N_Window – 1]

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- ◊ Y_In[0:N_Sp – 1] = Grid_Window[0:N_Window – 1]
- ◊ S = 0
- ◊ X_Int = Lat

And the output parameters of which are :

- ◊ Y_Int = Grid_Int[I_Lon]
- ◊ FP
- ◊ IFAIL

The following checks are made in case of Greenwhich meridian transition :

- For I_Lon = 1 to N_Window – 1
  - If Lon_Grid[I_Lon] – Lon_Grid[I_Lon – 1] < 0, then
    - Lon_Grid[I_Lon] = Lon_Grid[I_Lon] + Xcyc
- If Lon < Lon_Grid[0], then
  - Lon = Lon + Xcyc

Spline interpolation as function of longitude :

- Spline interpolate the grid values at the latitude of the altimeter measurement, using the mechanism "GEN_MEC_INT_07 – Spline interpolation",

  The input parameters of which are :

  - N_Sp = N_Window
  - X_In[0:N_Sp – 1] = Lon_Grid[0:N_Window – 1]
  - Y_In[0:N_Sp – 1] = Grid_Int[0:N_Window – 1]
  - S = 0
  - X_Int = Lon

  And the output parameters of which are :

  - Y_Int = Field_Int
  - FP
  - IFAIL

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_COM_01 - Arithmetic averaging

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J.P. DUMONT CLS	
Checked by:	For the JASON-1 SWT C.K. SHUM For the JASON-1 Project P. VINCENT CNES S. DESAI JPL For the F-PAC / ENVISAT Project J. BENVENISTE ESA P. VINCENT CNES	
Approved by:	P. VINCENT CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To compute the arithmetic mean value, the minimum value, the maximum value, the standard deviation and the RMS of a set of data.

ALGORITHM SPECIFICATION

Input data

- Number of points to be averaged : N (/)
- Point values : Val[0:N-1]

Output data

- Mean value : Mean (same unit as Val)
- Standard deviation [1] : Std (same unit as Val)
- Minimum value [1] : Min (same unit as Val)
- Maximum value [1] : Max (same unit as Val)
- Root mean square value [1] : Rms (same unit as Val)
- Execution status

Processing

$$\text{Mean} = \frac{\sum_j \text{Val}(j)}{N} \tag{1}$$

$$\text{Std} = \sqrt{\frac{\sum_j [\text{Val}(j) - \text{Mean}]^2}{N-1}} \tag{2}$$

$$\text{Rms} = \sqrt{\frac{\sum_j [\text{Val}(j)]^2}{N}} \tag{3}$$

Min and Max values are also computed.

[1] This output is generated on request

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

COMMENTS

The design of this algorithm will account for possible offsets to be able to compute Mean and Std even from a data set made of high values to avoid numerical problems.

REFERENCES

None

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_COM_02 - Linear regression / Least square method

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J.P. DUMONT CLS	
Checked by:	For the JASON-1 SWT C.K. SHUM For the JASON-1 Project P. VINCENT CNES S. DESAI JPL For the F-PAC / ENVISAT Project J. BENVENISTE ESA P. VINCENT CNES	
Approved by:	P. VINCENT CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To perform a linear regression (Least Square method) from a set of data, and to output the coefficients (order 1 and order 0) of the linear model and the standard deviation with respect to this model.

ALGORITHM SPECIFICATION

Input data

- Number of points of the regression : N (/)
- Set of ordinates : Y[0:N-1]
- Set of corresponding abscissa : X[0:N-1]

Output data

- Order 1 coefficient : C1 (same unit as Y/X)
- Order 0 coefficient : C0 (same unit as Y)
- Standard deviation with respect to the linear model [1] : Std (same unit as Y)
- Execution status

Processing

- Let D be defined by:

$$D = \left(\sum_i X(i)^2\right) * \left(\sum_i 1\right) - \left(\sum_i X(i)\right)^2 \quad (1)$$

- The order 1 coefficient C1 is computed by:

$$C1 = \frac{\left(\sum_i X(i) * Y(i)\right) * \left(\sum_i 1\right) - \left(\sum_i X(i)\right) * \left(\sum_i Y(i)\right)}{D} \quad (2)$$

- The order 0 coefficient C0 is computed by:

$$C0 = \frac{\left(\sum_i X(i)^2\right) * \left(\sum_i Y(i)\right) - \left(\sum_i X(i)\right) * \left(\sum_i X(i) * Y(i)\right)}{D} \quad (3)$$

[1] This output is generated on request

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- The standard deviation Std with respect to the linear model is computed by:

$$Std = \sqrt{\frac{\sum_i \left[Y(i) - (C1 * X(i) + C0)\right]^2}{N-2}} \quad (4)$$

COMMENTS

The design of this algorithm will account for possible offsets to be able to compute Std even from a data set made of high values to avoid numerical problems.

REFERENCES

None

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_COM_03 - Linear regression / Absolute deviation method

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J.P. DUMONT CLS	
Checked by:	For the JASON-1 SWT C.K. SHUM For the JASON-1 Project P. VINCENT CNES S. DESAI JPL For the F-PAC / ENVISAT Project J. BENVENISTE ESA P. VINCENT CNES	
Approved by:	P. VINCENT CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To perform a linear regression (Absolute Deviation method) from a set of data, and to output the coefficients (order 1 and order 0) of the linear model and the standard deviation with respect to this model.

ALGORITHM SPECIFICATION

Input data

- Number of points of the regression : N (/)
- Set of ordinates : Y[0:N-1]
- Set of corresponding abscissa : X[0:N-1]

Output data

- Order 1 coefficient : C1 (same unit as Y/X)
- Order 0 coefficient : C0 (same unit as Y)
- Standard deviation with respect to the linear model [1] : Std (same unit as Y)
- Execution status

Processing

The fitting of Y by minimizing absolute deviation is performed using the "medfit" function described in RD3.

The following notations are consistent with those used in RD3.

The "medfit" function is called with the following arguments:

x	= X (array of the ndata abscissa)	[input]
y	= Y (array of the ndata ordinates)	[input]
ndata	= N (number of points of the regression)	[input]
a	= C0 (order 0 coefficient)	[output]
b	= C1 (order 1 coefficient)	[output]
abdev	= Std (standard deviation with respect to the linear model)	[output]

Be aware of the following precision about the various functions requested to implement the linear regression by minimizing absolute deviation as proposed in RD3:

- The "medfit" function calls the "rofunc" function, which in turn calls the "select" function.

[1] This output is generated on request

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- The following updates of the functions described in RD3 are required:
  - Input arrays X and Y are defined for i = 0 to N-1, while arrays defined from 1 to N are expected by the "medfit" function
  - Types will be modified to conform with those used in the current application (if any)

COMMENTS

The design of this algorithm will account for possible offsets to be able to compute Std even from a data set made of high values to avoid numerical problems.

REFERENCES

None

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_COM_04 - Editing and compression

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J.P. DUMONT CLS	
Checked by:	For the JASON-1 SWT M. SROKOSZ E. RODRIGUEZ For the JASON-1 Project P. VINCENT CNES S. DESAI JPL For the F-PAC / ENVISAT Project J. BENVENISTE ESA P. VINCENT CNES	
Approved by:	P. VINCENT CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To compute compressed estimates from elementary estimates.

ALGORITHM SPECIFICATION

Input data

- Number of estimates to be compressed : N
- Estimates : X [0:N -1]
- Map of estimates to be compressed (1) : X_Comp_Flag [0:N-1] (/)
- Validity flags for the estimates (1) : X_Val_Flag [0:N-1] (/)
- Processing parameters:
  - Type of compression (2) : Type_Comp (/)
  - Minimum number of estimates requested for the compression : Min_Pts (/)
  - Minimum value of the standard deviation for outliers identification : Min_Std (same unit as X)
  - Standard deviation scale factor for outliers identification : Scale (/)
  - Center of the averaged measurement (3) : Center_Av_Meas (/)

Output data

- Compressed estimate : X_Mean (same unit as X)
- Standard deviation : X_Std (same unit as X)
- Map of valid estimates (1) : Map [0:N-1] (/)
- Number of valid estimates : Nval (/)
- Validity of the compressed estimate (flag) (1) : X_Mean_Val_Flag (/)
- Execution status

Processing

- If N=0 or if there is no measurement (j) such that the map of estimates to be compressed (X_Comp_Flag(j)) and the validity flag (X_Val_Flag(j)) are set to "valid", then:

  X_Mean and X_Std are set to default values

---

(1) 2 states : "valid" or "invalid"

(2) One state among the 3 following ones : "arithmetic averaging", "linear regression / absolute deviation method" or "linear regression / least square method"

(3) Not requested if Type_Comp is set to "Arithmetic averaging"

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- Nval is set to 0
- All the output flags (X_Mean_Val_Flag and flags of Map) are set to "invalid"

- Else:
  - The set of selected estimates (i.e. the set of estimates to be compressed) is first restricted to the estimates X(j) such as X_Comp_Flag(j) and X_Val_Flag(j) are set to "valid".
  - Then, the following iterative process (steps 1 to 3) is performed until Nval is constant ("normal ending"), or until it is stopped ("iterative process break").
    - Step 1 (Test of Nval):

      Let Nval be the current number of selected estimates.

      - If $Nval < Min_Pts$, then the iterative process is stopped ("iterative process break")
      - Else if Nval is unchanged with respect to the previous iteration (condition to be ignored for the first call), then the iterative process is stopped ("normal ending")
      - Else the following steps of the iterative process are performed.
    - Step 2 (Computation of X_Mean and X_Std):
      - If Type_Comp is set to "Arithmetic averaging", then the compressed estimate X_Mean and the standard deviation X_Std are computed using mechanism "GEN_MEC_COM_01 - Arithmetic averaging", with the following inputs:

        Number of points : The number Nval of selected estimates

        Points value : The Nval selected estimates {X(j)}
      - If Type_Comp is set to "Linear regression / Least square method", then the coefficients C1 and C0 (order 1 and order 0) of the linear model and the standard deviation X_Std are computed using mechanism "GEN_MEC_COM_02 - Linear regression / Least square method", with the following inputs:

        Number of points : The number Nval of selected estimates

        Ordinates : The Nval selected estimates {X(j)}

        Abscissa : The rank of the Nval corresponding measurements {j} ($j \in [0, N-1]$)

        The compressed estimate X_Mean is then derived from the two coefficients (C1 and C0) and from the center of the averaged measurement (Center_Av_Meas), by:

        $$X_Mean = C1 * Center_Av_Meas + C0 \quad (1)$$
      - If Type_Comp is set to "Linear regression / Absolute deviation method", then the coefficients C1 and C0 (order 1 and order 0) of the linear model and the standard deviation X_Std are computed using mechanism "GEN_MEC_COM_03 - Linear regression / Absolute deviation method", with the following inputs:

        Number of points : The number Nval of selected estimates

        Ordinates : The Nval selected estimates {X(j)}

        Abscissa : The rank of the Nval corresponding measurements {j} ($j \in [0, N-1]$)

        The compressed estimate X_Mean is then derived from the two coefficients (C1 and C0) and from the center of the averaged measurement (Center_Av_Meas), using (1).
    - Step 3 (Update of the selected estimates):

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- ◊ If $X_Std \leq Min_Std$, then the set of selected estimates is unchanged
- ◊ Else ($X_Std > Min_Std$) the set of selected estimates is restricted to the estimates X(j) such as:

$$|X(j)-X_Mod(j)| \leq Scale * X_Std \quad (2)$$

where:

X_Mod(j) = X_Mean if Type_Comp is set to "Arithmetic averaging"

X_Mod(j) = C1∗j + C0 if Type_Comp is set to "Linear regression / Absolute deviation method" or "Linear regression / Least Square method"

- ◊ Step 1 of the iterative process is then performed.

– Flags Map(j) of the map of valid estimates are set to "valid" for the estimates belonging to the last set of selected estimates, while they are set to "invalid" for the other estimates

– If the iterative process has a "normal ending", then the validity flag for the compressed estimate X_Mean_Val_Flag is set to "valid", else ("iterative process break") it is set to "invalid".

COMMENTS

None

REFERENCES

None

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_COM_05 - Estimation of the slope of the logarithm of the trailing edge of a waveform (linear regression)

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J.P. DUMONT CLS	
Checked by:	For the JASON-1 SWT G. HAYNE For the JASON-1 Project P. VINCENT CNES S. DESAI JPL For the F-PAC / ENVISAT Project J. BENVENISTE ESA P. VINCENT CNES	
Approved by:	P. VINCENT CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To estimate the slope of the logarithm of the trailing edge of a waveform, by linear regression (see comments for a brief mathematical statement).

ALGORITHM SPECIFICATION

Input data

- Window for the estimation:
  - Abscissa of the beginning : Abs_First (/)
  - Abscissa of the end : Abs_Last (/)
- Waveform:
  - Number of samples : Nb_Sample (/)
  - Abscissa of samples : Wf_Abs [0:Nb_Sample-1] (/)
  - Samples amplitude : Wf_Ampl [0:Nb_Sample-1] (FFT power unit)
  - Offset for normalization : Offset (FFT power unit)
  - Sampling interval of the analysis window : FFT_Step (s)

Output data

- Slope of the logarithm of the trailing edge : Slope (s^{-1})
- Flag for the computation of the slope of the trailing edge [1] : Flag_Slope (/)
- Execution status

Processing

- Flag_Slope is initialized to "valid".
- If the following condition is satisfied for less than two samples:

$$Abs_First \leq Wf_Abs(i) \leq Abs_Last \quad \text{and} \quad Wf_Ampl(i) - Offset > 0 \qquad (1)$$

  Then:
  - Flag_Slope is set to "invalid"
  - Slope is set to a default

  Else, Slope is computed as follows where the summation turns on i verifying (1), and where $Wf_Norm(i) = Wf_Ampl(i) - Offset$:

[1] 2 states: "valid" or "invalid"

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

$$\text{Slope} = \frac{1}{\text{FFT_Step}} * \frac{\sum_i 1 * \sum_i [\text{Wf_Abs(i)} * \log_e[\text{Wf_Norm(i)}]] - \sum_i \text{Wf_Abs(i)} * \sum_i \log_e[\text{Wf_Norm(i)}]}{\sum_i 1 * \sum_i \text{Wf_Abs(i)}^2 - \left(\sum_i \text{Wf_Abs(i)}\right)^2} \quad (2)$$

COMMENTS

Ignoring the skewness effects, the expression versus time of the trailing edge of the Hayne's model (Hayne, 1980) may be represented as follows (as described in AD4: "ALT_RET_ICE_02 - To perform the ice-2 retracking" or "ALT_RET_OCE_02 - To perform the ocean-2 retracking"):

$$\text{Vm}(t) = a_\xi P_u \exp(-v) + P_n \quad (1)$$

where: P_u is the amplitude of the model, P_n is the thermal noise level

$$a_\xi = \exp\left(\frac{-4\sin^2\xi}{\gamma}\right), \quad \gamma = \frac{2}{\text{Log}_e(2)}.\sin^2\left(\frac{\theta_0}{2}\right)$$

ξ is the off-nadir angle and θ_O is the antenna beamwidth

$$v = c_\xi\left(t - \tau - \frac{c_\xi \sigma_c^2}{2}\right), \quad c_\xi = \alpha * \left[\cos(2\xi) - \frac{\sin^2(2\xi)}{\gamma}\right], \quad \alpha = \frac{4c}{\gamma h\left(1 + \frac{h}{R_e}\right)}$$

τ is the epoch of the model, c is the light velocity, h is the satellite altitude and R_e is the earth radius

The following expression may be derived from (1):

$$\text{Log}_e[\text{Vm}(t) - P_n] = -c_\xi t + \left(c_\xi \tau + \frac{c_\xi^2 \sigma_c^2}{2} + \log_e(a_\xi P_u)\right) \quad (2)$$

It will thus be possible to derive an estimate of the off-nadir angle (ξ) from the slope of the logarithm of the trailing edge ($-c_\xi$). This slope may be estimated by linear regression of the logarithm of the waveform samples (from which the thermal noise level estimate is removed), in a predefined gate set on the trailing edge.

REFERENCES

- Hayne G.S. 1980: "Radar Altimeter Mean Return Waveforms from Near-Normal-Incidence Ocean Surface Scattering". IEEE Trans. on antennas and propagation, Vol. AP-28, n°5, pp. 687-692

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_COM_06 - Flag compression

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	P.THIBAUT	CLS	
Checked by:	N. PICOT	CNES	
Approved by:	P. VINCENT	CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	Date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To compute a flag that sums up a set of flags

ALGORITHM SPECIFICATION

Input data

- Number of flags to check : Num_Flag (/)
- Flags to check : Flag_In [0 :Num_Flag-1] (/) (1)

Output data

- Flag that sums up the input flags : Flag_Out (/) (1)

Processing

This algorithm is aimed at providing a general flag that sums up a set of flags:

- If all the Flag_In are set to “invalid” then Flag_Out is set to “invalid”.
- If all the Flag_In are set to “valid” the Flag_Out is set to “valid”
- If at least one Flag_In is set to “invalid”, the Flag_Out is set to “invalid”

COMMENTS

None

REFERENCES

None

(1) 2 states: "valid" or "invalid"

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_COM_07 - Polynomial regression / Least Square method

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	S. LABROUE	CLS	
Checked by:	N. PICOT	CNES	
Approved by:	P. VINCENT	CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To perform polynomial regression on a set of measurements, using a Least Square fit and a QR factorization.

ALGORITHM SPECIFICATION

Input data

- Number of measurements : N (/)
- Set of measurements : Mes [0:N-1]
- Number of polynomial coefficients : N_pol (/)
- Matrix of the polynomial regression : B [0:N-1] [0:N_pol-1]

Output data

- Adjusted measurements : Mes_mod [0:N-1] (same unit as Mes)
- Residuals on the measurements : Res [0:N-1] (same unit as Mes)
- Standard deviation of the residuals : Std_res (same unit as Mes)

Processing

- To perform the QR decomposition of the input matrix B :
  - To initialize the matrix R, the right hand side T and the quality index S_qual, before starting the decomposition :

$$R = 0 \quad (1)$$

$$T = 0 \quad (2)$$

$$S_qual = 0 \quad (3)$$

  - For each row of the input matrix, i = 0, N-1

    To update matrix R and right hand side T, using mechanism “GEN_MEC_MAT_01 - Updating a QR decomposition using Givens rotation ” with the following inputs :

    * Number of elements of the matrix row : N_pol (/)
    * Row of the matrix B : B[i] [0:N_pol-1]
    * Index of the first non zero coefficient of the row : 0 (/)
    * Component of the right hand side corresponding to the matrix row : Mes(i)
    * Matrix R to update (triangular but stored in a vector row by row) : R[0: N_pol*(N_pol+1)/2 –1]
    * Right hand side T to update : T[0:N_pol-1]
    * Quality index of the LS method to update : S_qual

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

The outputs are the updated matrix R, right hand side T and quality index S_qual

- To solve the triangular system RX = T

  Vector X solution of system R X = T is computed using routine F06PLF of the NAG Fortran library.

  This function is called with the following arguments:

UPLO	= 'L'	[input]
TRANS	= 'T'	[input]
DIAG	= 'N'	[input]
N	= N_pol	[input]
AP	= R (triangular matrix stored in one vector)	[input]
X	= T	[input/output]
INCX	= 1	[input]
LENGTH_1	= 1 (length of the string UPLO)	[input]
LENGTH_2	= 1 (length of the string TRANS)	[input]
LENGTH_3	= 1 (length of the string DIAG)	[input]

  The output (the solution X) is stored in the vector T

- To compute the adjusted measurements by performing the product Y = B*X

  Vector Y is computed using routine F06PAF of the NAG Fortran library.

  This function is called with the following arguments:

TRANS	= 'N'	[input]
M	= N	[input]
N	= N_pol	[input]
ALPHA	= 1	[input]
A	= B (rectangular matrix)	[input]
LDA	= N (first dimension of the array B as declared in the calling program)	[input]
X	= T	[input]
INCX	= 1	[input]
BETA	= 0	[input]
Y	= Y (Y = 0 in input, Y = B*X in output)	[input/output]
INCY	= 1	[input]
LENGTH_1	= 1 (length of the string TRANS)	[input]

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

The output is stored in the vector Y

- To store the adjusted measurements :

$$Mes_mod = Y \tag{4}$$

- To compute the standard deviation on the residuals :

$$Res = Mes - Y \tag{5}$$

$$Std_res = \sqrt{\frac{\sum_{i=0}^{N_pol-1} Res(i)^2}{N_pol - 1}} \tag{6}$$

COMMENTS

None

REFERENCES

None

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_COM_08 - Least-squares fit by polynomial or periodic functions

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J. STUM	CLS	
Checked by:	N. PICOT	CNES	
Approved by:	P. VINCENT	CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	Date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To compute a least square fit of a Y series versus an X series by using polynomial or periodic functions. Variances of the Y values are provided in input for proper weighting in the least-squares fit procedure. The least square fit leads to solve a symmetric indefinite system of linear equations AX = B, where X is a column matrix which contains the unknowns (coefficients of the polynomials or periodic functions), A is the so-called curvature matrix and is a symmetric square matrix, and B is the right-hand side column matrix. Bunch_Kaufman factorization of A is done first, using the F07MDF routine of the Numerical Algorithms Group (NAG) library: $A = P.L.D.L^T.P^T$, where P is a permutation matrix, L is a unit lower triangular matrix, and D is a symmetric block diagonal matrix with 1 by 1 and 2 by 2 diagonal blocks. System resolution is then done using the F07MEF routine of NAG library.

ALGORITHM SPECIFICATION

Input data

- The number of points : NP(/)
- The abscissas : X[0:NP-1]
- The ordinates : Y[0:NP-1]
- The variances of the ordinates : Var[0:NP-1]
- The type of fit (= 'pol' for polynomials, = 'sin' for periodic functions) : Type_Fit
- The degree of polynomial fit (if Type_Fit = 'pol'), or
  the number of frequencies to be fitted (if Type_Fit = 'sin') : Degree (/)
- The frequencies to be fitted : Freq[0:Degree-1] (s^{-1})

Output data

- The adjusted ordinates : Y_Adj[0:NP-1]
- The coefficients of the polynomial or periodic fit : B[0:2∗Degree]

Processing

- For I = 0 to NP – 1
  - If Type_Fit = 'pol', then :
    - For M = 0 to Degree
      - ◊ $F[M][I] = (X[I])^M$ (1)
  - Else (Type_Fit = 'sin') :
    - $F[0][I] = 1$ (2)
    - $J = 1$ (3)
    - For M = 0 to Degree – 1
      - ◊ $F[J][I] = \sin(2\pi * Freq[M] * X[I])$ (4)

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- ◊ J = J+1 (5)
- ◊ F[J][I] = cos (2π ∗ Freq[M] ∗ X[I]) (6)
- ◊ J = J+1 (7)

- If Type_Fit = 'pol', then
  - Order = Degree + 1 (8)
- Else (Type_Fit = 'sin') :
  - Order = 2∗Degree + 1 (9)

Initialization of the matrices A and B:

- For M = 0 to Order – 1
  - B[M] = 0 (10)
  - For K = M to Order – 1
    - ∗ A[M][K] = 0 (11)

Calculation of the elements of the matrix A :

- For M = 0 to Order – 1
  - For K = M to Order – 1
    - ∗ For I = 0 to NP – 1
      - ◊ $A[M][K] = A[M][K] + \frac{F[M][I] * F[K][I]}{Var[I]}$ (12)
    - ∗ A[K][M] = A[M][K] (13)

Calculation of the elements of the matrix B :

- For M = 0 to Order – 1
  - For I = 0 to NP – 1
    - ∗ $B[M]= B[M] + \frac{Y[I] * F[M][I]}{Var[I]}$ (14)

- The matrix A is first Bunch-Kaufmann factorized, using the routine F07MDF of the NAG library, the parameters of which are:
  - UPLO (input) = 'L'
  - N (input) = Order
  - A (input/ouput) = A[0:Order-1][0:Order-1]
  - LDA (input) = Order
  - IPIV (output) : array of Order points
  - WORK (output) : Unused (array of LWORK points)
  - LWORK (input) = 2∗Order
  - INFO (output)

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- The system AX = B is then solved, using the routine F07MEF of the NAG library, the parameters of which are:
  - UPLO (input) = 'L'
  - N (input) = Order
  - NRHS (input) = 1
  - A (input) = A[0:Order-1][0:Order-1] (output of F07MDF)
  - LDA (input) = Order
  - IPIV (input) : output of F07MDF
  - B (input/output) = B[0:Order-1]
  - LDB (input) = Order
  - INFO (output)

Calculation of the output adjusted ordinates :

- For I = 0 to NP – 1
  - Y_Adj[I] = 0 (15)
  - For M = 0 to Order – 1
    - Y_Adj[I] = Y_Adj[I] + (B[M] ∗ (F[M][I]) (16)

COMMENTS

None

REFERENCES

Bevington, P. R. : Data Reduction and Error Analysis for the Physical Sciences, Mc Graw-Hill, 1969.

See also RD8.

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_COM_09 - Multiple linear regression

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J. STUM	CLS	
Checked by:	N. PICOT	CNES	
Approved by:	P. VINCENT	CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	Date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To perform multiple linear regression between a variable Y (the dependent variable) and a set of variables X_i (the independent variables) : $Y = A + \sum_i B_i * X_i$.

ALGORITHM SPECIFICATION

Input data

- The number of variables of the regression (sum of the Y variable and the X variables) : K1 (/)
- The number of measurements : N (/)
- The variables of the regression : X[0:K1-1][0:N-1][1]

Output data

- The regression constant : A (same unit as Y)
- The regression coefficients : B[0:K1-2] (same unit as Y/X_i)
- The execution status

Processing

- For I_Var = 0 to K1-1 :
  - The mean value Mean_Var[I_Var] and standard deviation Std_Var[I_Var] of the variable X[I_Var][0:N-1] are computed, using the mechanism "GEN_MEC_COM_01 – Arithmetic averaging",

    The input parameters of which are :

    * N
    * Val[0:N-1] = X[I_Var][0:N – 1]

    And the output parameters of which are :

    * Mean = Mean_Var[I_Var]
    * Std = Std_Var[I_Var]
    * Min (unused)
    * Max (unused)
    * RMS (unused)
    * The execution status

[1] The dependent variable Y is the last one in array X (Y = X[K1-1][0 :N-1])

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- For J_Var = 0 to K1-1 :
  - The mean value Mean_Prod_Var[I_Var][J_Var] of the product of the two variables X[I_Var][0:N-1] and X[J_Var][0:N-1] is computed :

$$\text{Mean_Prod_Var[I_Var][J_Var]} = \frac{\sum_{I=0}^{N-1} X[I_Var][I] * X[J_Var][I]}{N} \quad (1)$$

- For I_Var = 0 to K1-1
  - For J_Var = 0 to K1-1

    The covariance and the correlation factor of the two variables I_Var and J_Var are computed :
    - Covar[I_Var][J_Var] = Mean_Prod_Var[I_Var][J_Var] - Mean_Var[I_Var]∗Mean_Var[J_Var] (2)
    - If I_Var = J_Var, then :
      - ◊ Correl[I_Var][J_Var] = 1 (3)
    - Else :
      - ◊ $\text{Correl[I_Var][J_Var]} = \frac{1}{\sqrt{\frac{N-1}{N}}} \frac{\text{Covar[I_Var][J_Var]}}{\text{Std_Var[I_Var]} * \text{Std_Var[J_Var]}}$ (4)

- The regression constant A and the regression coefficients B[0:K1-2] are computed, using the routine G02CGF of the NAG library,

  the parameters of which are:
  - N (input)
  - K1 (input)
  - K (input) = K1-1
  - XBAR (input) = Mean_Var[0:K1-1]
  - SSP (input) = N∗Covar[0:K1-1][0:K1-1]
  - ISSP (input) = K1
  - R (input) = Correl[0:K1-1][0:K1-1]
  - IR (input) = K1
  - RESULT(13) (output) : Unused (real array of 13 elements)
  - COEFF[0:ICOEFF-1][0:2] (output) : contains the regression coefficients and statistics
  - ICOEFF (input) = K1-1
  - CONST[0:2]) (output) : contains the regression constants and statistics
  - RINV[0:IRINV-1][0:K-1] (output) : Unused (real matrix)
  - IRINV (input) = K1-1
  - C[0:IC-1][0:K-1] (output) : Unused (real matrix)
  - IC (input) = K1-1

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- WKZ[0:IWKZ-1][0:K-1] (output) : Unused (real matrix)
- IWKZ (input) = K1-1
- IFAIL (input/output) = -1 on input

- A = CONST[0] (5)
- For I_Var = 0 to K1-2
  - B[I_Var] = COEFF[I_Var][0] (6)

COMMENTS

None

REFERENCES

None

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_QUA_01 - Quality check from thresholds

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J.P. DUMONT CLS	
Checked by:	For the JASON-1 SWT C.K. SHUM For the JASON-1 Project P. VINCENT CNES S. DESAI JPL For the F-PAC / ENVISAT Project J. BENVENISTE ESA P. VINCENT CNES	
Approved by:	P. VINCENT CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To check the quality of a parameter versus two thresholds and to provide a 2-state validity flag.

ALGORITHM SPECIFICATION

Input data

- Parameter to be checked : X
- Validity thresholds:
  - Lower bound : Min_X (same unit as X)
  - Upper bound : Max_X (same unit as X)

Output data

- Validity flag for X [(1)] : X_Val_Flag (/)

Processing

- If $Min_X \leq X \leq Max_X$, then: X_Val_Flag is set to "valid"
- Else: X_Val_Flag is set to "invalid"

COMMENTS

None

REFERENCES

None

[(1)] 2 states: "valid" or "invalid"

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_QUA_03 - Editing of measurements against thresholds

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J. STUM	CLS	
Checked by:	N. PICOT	CNES	
Approved by:	P. VINCENT	CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To edit the measurement for which the value of some particular field (or linear combination of some particular fields) is outside provided threshold values. The list of fields to be checked is given in the input static auxiliary data.

ALGORITHM SPECIFICATION

Input data

- Number of measurements : Nb_Mes (/)
- Number of fields within a measurement : N_Fields (/)
- Measurements : Meas[0:Nb_Mes-1][0:N_Fields-1]
- Validity flags of the measurements : VF[0:Nb_Mes-1] (/) (1)
- Number of parameters within a measurement to be checked : Nb_Param (/)
- For each parameter to be checked (For I_Param = 0 to Nb_Param-1):
  - Number of fields within the measurement, from which the parameter is computed : Nb_Fields (/)
  - For each field (for I_Field = 0 to Nb_Fields-1)
    - Field number from which the parameter is computed : Num_Field (/)
    - Sign of each field when computing the parameter from their linear combination : Sign (2)
  - Lower threshold value : Thresh_Inf (same unit as Meas)
  - Upper threshold value : Thresh_Sup (same unit as Meas)

Output data

- Validity flags of the measurements : VF[0:Nb_Mes-1] (/) (1)
- Total number of measurements having VF set to “valid” in input : Nb_Good (/)
- Total number of measurements having VF set to “invalid” in output : Nb_Bad (/)
- For each parameter checked (For I_Param = 0 to Nb_Param-1)
  - Number of measurements having VF set to “invalid” in output : Nb_Bad_Param (/)
- Execution status

(1) 2 states : “valid” and “invalid”

(2) “plus” or “minus”

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Processing

- Nb_Good = 0 (1)
- Nb_Bad = 0 (2)
- For each parameter to be checked (For I_Param = 0 to Nb_Param-1):

  Nb_Bad_Param[I_Param] = 0 (3)
- For each input measurement (for I_Mes = 0 to Nb_Mes-1):
  - If VF(I_Mes) is set to “valid”, then:
    - ∗ Nb_Good = Nb_Good + 1 (4)
    - ∗ For each parameter to be checked (For I_Param = 0 to Nb_Param-1):
      - ◊ VF_Param[I_Param] is set to “valid”
      - ◊ Val_Param[I_Param] = 0 (5)
      - ◊ For each field within the measurement, from which the parameter is computed (For I_Field = 0 to Nb_Fields – 1)
        - ⇒ If Meas[I_Mes][Num_Field[I_Param][I_Field]] is not set to its default value, then :
          - → Val_Field = Sign[I_Param][I_Field] ∗ Meas[I_Mes][Num_Field[I_Param][I_Field]] (6)
          - → Val_Param[I_Param] = Val_Param[I_Param] + Val_Field (7)
        - ⇒ Else :
          - → VF_Param[I_Param] is set to “invalid”
      - ◊ Valmin = Thresh_Inf[I_Param] (8)
      - ◊ Valmax = Thresh_Sup[I_Param] (9)
      - ◊ If VF_Param[I_Param] is not set to “invalid”, then :
        - ⇒ The validity flag VF(I_Mes) is determined, using mechanism “GEN_MEC_QUA_01 - Quality check from thresholds”,

          the input parameters of which are:

          → X = Val_Param[I_Param]

          → Min_X = Valmin

          → Max_X = Valmax

          and the output parameter of which is:

          → X_Val_Flag = VF_Param(I_Param)
      - ◊ If VF_Param(I_Param) is set to “invalid” then

        Nb_Bad_Param[I_Param] = Nb_Bad_Param[I_Param] + 1 (10)

        VF[I_Mes] is set to “invalid”
    - ∗ If VF(I_Mes) is set to “invalid”, then:

      Nb_Bad = Nb_Bad + 1 (11)

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

COMMENTS

None

REFERENCES

None

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_QUA_04 - Check of a parameter versus a reference value

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	P. THIBAUT CLS	
Checked by:	For the JASON-1 SWT D. HANCOCK For the JASON-1 Project P. VINCENT CNES S. DESAI JPL For the F-PAC / ENVISAT Project J. BENVENISTE ESA P. VINCENT CNES	
Approved by:	P. VINCENT CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To check a parameter versus a reference value.

ALGORITHM SPECIFICATION

Input data

- Parameter to be checked : X
- Reference value of the parameter : X_Ref (same unit as X)
- Threshold percentage : Pcent_X (/)

Output data

- Validity flag for X [1] : Flag_X (/)

Processing

The validity flag Flag_X is set to “invalid” at the beginning of the procedure.

- If X_Ref=0 or X=0 then Flag_X is set to “invalid”
- $Diff_Par = ABS[X_Ref - X]$ (1)
- If $Diff_Par > 0$ then: (2)
  - If $ABS(X) > ABS(X_Ref)$ then: $Diff_Pcent = \frac{Diff_Par}{ABS(X_Ref)} * 100$ (3)

    Else: $Diff_Pcent = \frac{Diff_Par}{ABS(X)} * 100$ (4)
  - If $Diff_Pcent \leq Pcent_X$ then Flag_X is set to "valid" (5)

  Else Flag_X is set to "valid".

COMMENTS

If the input parameter X is set to NaN, then flag_X in output must be set to “invalid”.

[1] 2 states: "valid" or "invalid"

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

REFERENCES

None

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_QUA_05 - Editing of measurements using spline

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J. STUM	CLS	
Checked by:	N. PICOT	CNES	
Approved by:	P. VINCENT	CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To edit measurements for which the difference between the sea surface height and a corresponding spline-fitted sea surface height exceeds some threshold value.

ALGORITHM SPECIFICATION

Input data

- Number of measurements : NM (/)
- Sea surface height measurements : SSH[0:NM-1] (m)
- Measurements time tags : Time_Tag[0:NM-1][1]
- The validity flags of the measurements : VF[0:NM-1] (/)
- The time interval between two consecutive measurements : Delta_Time (s)
- The number of points used to compute the cubic spline (even) : N_Sp (/)
- The noise on SSH : Noise (m)
- The stopping criteria : SC (m)
- The percentage of correction of SSH : PER (/)
- The threshold on SSH : Thresh_SSH (m)

Output data

- The validity flags of the measurements : VF[0:NM-1][2] (/)
- The number of measurements having VF set to “valid” in input : Nb_Good (/)
- The number of measurements having VF set to “invalid” in output : Nb_Bad (/)

Processing

- Nb_Good = 0 (1)

Computing the smoothing factor for the E02BEF NAG routine from the SSH noise :

- SF = (N_Sp – 2)∗Noise∗Noise (2)
- For each input measurement : (For I_Mes = 0 to NM-1) :
  - If VF[I_Mes] is set to “valid”, then :
    - Num_Mes[Nb_Good] = I_Mes (3)

---

(1) seconds elapsed since 1/1/1950 0h

(2) 2 states : « valid » or « invalid »

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- If Nb_Good = 0, then
  - Time_Tag_First = Time_Tag[I_Mes] (4)
  - Time_Tag_Rel[Nb_Good] = 0 (5)
- Else
  - Time_Tag_Rel[Nb_Good] = Time_Tag[I_Mes] – Time_Tag_First (6)
- Nb_Good = Nb_Good + 1

- If Nb_Good ≤ N_Sp, then
  - Nb_Bad = 0 (7)
  - VF[0:NM-1] in output are set to their values in input. (8)
  - The processing hereafter is not performed
- For each input valid measurement (For I_Val = 0 to Nb_Good-1), the 3 following processing steps are performed:
  - Index_First = 0 (9)
  - Index_Last = 0 (10)

  Step 1 :
  - N = N_Sp+1 measurements are selected, I_Val being the (N_Sp/2 +1)th measurement, using the mechanism "GEN_MEC_SEL_01 - Selection of N points for spline calculation",

    The input parameters of which are:
    - Index_Higher = I_Val
    - Index_Min = 0
    - Index_Max = Nb_Good – 1
    - N = N_Sp+1
    - Time[Index_Min:Index_Max] = Time_Tag_Rel[0:Nb_Good-1]
    - Delta_Time

    And the output parameters of which are :
    - Index_First
    - Index_Last
    - Flag_Avail = Avail(I_Val)
    - The execution status
  - If Avail(I_Val) = 2, then :
    - Delta_SSH(I_Val) = 0 (11)
    - Do not perform the next steps and process next point I_Val
  - Nb_Meas_Before[I_Val] = I_Val – Index_First (12)
  - Nb_Meas_After[I_Val] = Index_Last – I_Val (13)

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Step 2 :

- Consider the Nb_Meas_Before measurements before point I_Val and the Nb_Meas_After measurements after point I_Val (without point I_Val), to build the N_Sp input points of the mechanism "GEN_MEC_INT_07 - Spline interpolation" :
  - Nb_Meas_Shift = – Nb_Meas_Before[I_Val] (14)
  - For J = 0 to N_Sp – 1
    - X_In[J] = Time_Tag_Rel[I_Val + Nb_Meas_Shift] (15)
    - Y_In[J] = SSH[Num_Mes[I_Val + Nb_Meas_Shift]] (16)
    - Nb_Meas_Shift = Nb_Meas_Shift + 1 (17)
    - If Nb_Meas_Shift = 0, then

      ⇒ Nb_Meas_Shift = 1 (18)
- The SSH at point I_Val is interpolated by a spline computed on this set of N_Sp points, using the mechanism "GEN_MEC_INT_07 - Spline interpolation",

  The input parameters of which are :
  - N_Sp
  - X_In[0:N_Sp-1]
  - Y_In[0:N_Sp-1]
  - S = SF
  - X_Int = Time_Tag_Rel[I_Val]

  And the output parameters of which are :
  - Y_Int = SSH_Int
  - FP
  - IFAIL
  - The execution status

Step 3 :

- The difference between the SSH and the spline at measurement is computed :

  Delta_SSH[I_Val] = SSH[Num_Mes[I_Val]] – SSH_Int (19)

- Nb_Iter = 0 (20)
- While the maximum number of iterations is not overpassed, (Nb_Iter ≤ 100), do the following processing :
  - The measurements for which ABS(Delta_SSH) > SC are selected (if no such measurement exists, the processing hereafter is not performed).
  - Among them, the measurement having the maximum difference (ABS(Delta_SSH) – SC) is searched. Let I_Max be the index of this measurement.

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- Its SSH is corrected as follows :
  - SSH_Cor = SSH – (Delta_SSH x Per), where Per is a percentage of correction, $0 \leq Per \leq 1$ (21)
- Nb_Iter = Nb_Iter + 1 (22)
- The splines which have been computed using the measurement point which index is I_Max are recomputed using its corrected SSH, SSH_Cor :
  - For I_Val = I_Max – Nb_Meas_Before[I_Max] to I_Max + Nb_Meas_After[I_Max]
    - If Nb_Meas_Before[I_Val] $\neq$ 0 AND Nb_Meas_After[I_Val] $\neq$ 0, then :
      - $\Rightarrow$ Steps number 2 and 3 above are rerun

- Nb_Bad = 0 (23)
- For each input valid measurement : (for I_Val = 0 to Nb_Good-1) :
  - If ((I_Val = 0) OR (I_Val = Nb_Good – 1) OR (Avail(I_Val) = 2)), then
    - VF[Num_Mes[I_Val]] is set to “valid” (24)
  - Else :
    - If ABS[SSH_Cor[I_Val] – SSH[I_Val]] $\leq$ Thresh_SSH, then
      - VF[Num_Mes[I_Val]] is set to “valid” (25)
    - Else
      - VF[Num_Mes[I_Val]] is set to “invalid”
      - Nb_Bad = Nb_Bad + 1 (26)

COMMENTS

None

REFERENCES

None

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_COR_01 - Computation of a tracking/retracking combined parameter

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J.P. DUMONT CLS	
Checked by:	For the JASON-1 SWT M. SROKOSZ E. RODRIGUEZ For the JASON-1 Project P. VINCENT CNES S. DESAI JPL For the F-PAC / ENVISAT Project J. BENVENISTE ESA P. VINCENT CNES	
Approved by:	P. VINCENT CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To combine tracking-derived parameters (tracker range or scaling factor for Sigma0 evaluation) and retracking-derived parameters (epoch or amplitude of the waveform). The combined parameter is either the altimeter range or the backscatter coefficient.

ALGORITHM SPECIFICATION

Input data

- Tracking-derived parameter : Track_Par [1]
- Tracking-derived parameter validity flag [2] : Track_Par_Val_Flag (/)
- Retracking-derived parameter : Retrack_Par [1]
- Retracking-derived parameter validity flag : Retrack_Par_Val_Flag (/)
- Type of conversion for the retracking-derived parameter [3] : Type_Conv (/)
- Light velocity : Light_Vel (m/s)

Output data

- Tracking/retracking combined parameter : Comb_Par [1]
- Tracking/retracking combined parameter validity flag [2] : Comb_Par_Val_Flag (/)
- Execution status

Processing

- If the validity flags for the tracking-derived parameter (Track_Par_Val_Flag) and for the retracking-derived parameter (Retrack_Par_Val_Flag) are set to "valid", then:
  - The retracking-derived parameter is converted as follows:
    - If Type_Conv is set to "second to meter", then:

$$\text{Retrack_Par_Conv} = \frac{\text{Light_Vel}}{2} * \text{Retrack_Par} \qquad (1)$$

---

[1] The tracking-derived parameter and the converted retracking-derived parameter must have the same unit, which will be the unit of the combined parameter

[2] 2 states: "valid" or "invalid"

[3] "second to meter", "power to dB" or "none". If Type_Conv is set to "second to meter" then Track_Par and Comb_Par have units of meter and Retrack_Par has units of second. If Type_Conv is set to "power to dB" then Track_Par and Comb_Par have units of dB and Retrack_Par has units of power.

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- Else if Type_Conv is set to "power to dB", then:
  - If $Retrack_Par > 0$, then:

$$Retrack_Par_Conv = 10 * \log_{10}(Retrack_Par) \quad (2)$$

  - Else ($Retrack_Par \leq 0$) [4]:

    Comb_Par is set to a default value

    Comb_Par_Val_Flag is set to "invalid"

    The process is finished
- Else (Type_Conv set to "none"):

$$Retrack_Par_Conv = Retrack_Par \quad (3)$$

- The tracking/retracking combined parameter is computed by:

$$Comb_Par = Track_Par + Retrack_Par_Conv \quad (4)$$

- The corresponding validity flag (Comb_Par_Val_Flag) is set to "valid"

- Else (Track_Par_Val_Flag or Retrack_Par_Val_Flag set to "invalid"):
  - Comb_Par is set to a default value
  - The corresponding validity flag (Comb_Par_Val_Flag) is set to "invalid"

COMMENTS

None

REFERENCES

None

[4] The combined parameter is computed only if its nominal formulation can be applied. It is set to a default value and the associated validity flag is set to "invalid" otherwise

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_COR_02 - Computation of a corrected parameter from its raw value and its corrections

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J.P. DUMONT CLS	
Checked by:	For the JASON-1 SWT G. HAYNE E. RODRIGUEZ For the JASON-1 Project P. VINCENT CNES S. DESAI JPL For the F-PAC / ENVISAT Project J. BENVENISTE ESA P. VINCENT CNES	
Approved by:	P. VINCENT CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To compute a corrected value of a parameter from its raw value and its corrections. This algorithm sums a raw parameter and its various corrections.

ALGORITHM SPECIFICATION

Input data

- Raw value : X_Raw
- Validity flag of the raw value [1] : X_Raw_Val_Flag (/)
- Number of corrections : Nb_Cor (/)
- Corrections : Cor [0:Nb_Cor-1] (same unit as X_Raw)

Output data

- Corrected value : X_Cor (same unit as X_Raw)
- Execution status

Processing

- If X_Raw_Val_Flag is set to "valid", then:

$$X_Cor = X_Raw + \sum_{j=0}^{Nb_Cor-1} Cor(j) \qquad (1)$$

- Else (X_Raw_Val_Flag set to "invalid"): X_Cor is set to a default value

COMMENTS

None

REFERENCES

None

[1] 2 states: "valid" or "invalid"

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_COR_03 - Computation of the Doppler correction on the altimeter range

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J.P. DUMONT CLS	
Checked by:	For the JASON-1 SWT D. HANCOCK P. CALLAHAN F. LEMOINE For the JASON-1 Project P. VINCENT CNES S. DESAI JPL For the F-PAC / ENVISAT Project J. BENVENISTE ESA P. VINCENT CNES	
Approved by:	P. VINCENT CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To compute the Doppler correction on the altimeter range.

ALGORITHM SPECIFICATION

Input data

- Altitude rate with respect to the surface [(1)] : Alt_Rate (m/s)
- Instrumental characterization data:
  - Emitted frequency : Frequency (Hz)
  - Pulse duration : Pulse_Duration (s)
  - Emitted bandwidth : Bandwidth (Hz)
  - Sign of the slope of the transmitted chirp : Sign_Slope (-1 or +1)

Output data

- Doppler correction : Cor_Dop_Range (m)
- Execution status

Processing

The Doppler correction is computed by:

$$Cor_Dop_Range = -Sign_Slope * \frac{Frequency * Pulse_Duration}{Bandwidth} * Alt_Rate \quad (1)$$

COMMENTS

None

REFERENCES

None

---

(1) Ellipsoid for the OSDR parameters, MSS/Geoid for the JASON-1 IGDR/GDR parameters

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_COR_04 - Computation of a modeled instrumental correction

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J.P. DUMONT CLS	
Checked by:	For the JASON-1 SWT G. HAYNE For the JASON-1 Project P. VINCENT CNES S. DESAI JPL For the F-PAC / ENVISAT Project J. BENVENISTE ESA P. VINCENT CNES	
Approved by:	P. VINCENT CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To compute the modeled correction of instrumental errors for an altimetric parameter.

ALGORITHM SPECIFICATION

Input data

- Significant waveheight and validity flag:
  - Significant waveheight : SWH (m)
  - Validity flag for the significant waveheight [1] : SWH_Val_Flag (/)
- Signal to noise ratio and validity flag:
  - Signal to noise ratio : SNR (dB)
  - Validity flag for the signal to noise ratio [1] : SNR_Val_Flag (/)
- Mispointing correction on the backscatter coefficient : Cor_Mis_Sigma0 (dB)
- Default value for the inputs of the correction table:
  - SWH default value : SWH_Def (m)
  - SNR default value : SNR_Def (dB)
- Modeled instrumental correction table:
  - SWH lower bound : Min_SWH (m)
  - SWH upper bound : Max_SWH (m)
  - SWH step : Step_SWH (m)
  - SNR lower bound : Min_SNR (dB)
  - SNR upper bound : Max_SNR (dB)
  - SNR step : Step_SNR (dB)
  - Correction table : Tab [0:N_SWH-1][0:N_SNR-1] [2]
    with N_a=(Max_a-Min_a)/Step_a+1

Output data

- Modeled instrumental correction : Cor_Mod [2]
- Execution status

---

[1] 2 states: "valid" or "invalid"

[2] Same unit as the unit of the parameter for which the correction is computed (m for ranges and waveheights, m/s for range rates, dB for backscatter coefficients)

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Processing

- The interpolation versus the significant waveheight is prepared as follows:
  - If SWH_Val_Flag is set to "invalid", then the significant waveheight is set to its default value:

$$SWH = SWH_Def \tag{1}$$

  - The significant waveheight is limited and indexes ISWH_Prev and ISWH_Next are computed as follows:
    - If $SWH < Min_SWH$, then: $SWH = Min_SWH$ (2)
    - If $SWH < Max_SWH$, then:

$$ISWH_Prev = INT\left[\frac{SWH - Min_SWH}{Step_SWH}\right] \tag{3}$$

Else ($SWH \geq Max_SWH$), then:

$$SWH = Max_SWH \tag{4}$$

$$ISWH_Prev = INT\left[\frac{SWH - Min_SWH}{Step_SWH}\right] - 1 \tag{5}$$

    - $ISWH_Next = ISWH_Prev + 1$ (6)
  - The corresponding significant waveheights are computed by:
    - $SWH_Prev = Min_SWH + ISWH_Prev * Step_SWH$ (7)
    - $SWH_Next = SWH_Prev + Step_SWH$ (8)
- The interpolation versus the signal to noise ratio is prepared as follows:
  - If SNR_Val_Flag is set to "invalid", then the signal to noise ratio is set to its default value:

$$SNR = SNR_Def \tag{9}$$

  - The signal to noise ratio is corrected for the mispointing effects, by:

$$SNR = SNR + Cor_Mis_Sigma0 \tag{10}$$

  - The corrected signal to noise ratio is limited and indexes ISNR_Prev and ISNR_Next are computed as follows:
    - If $SNR < Min_SNR$, then: $SNR = Min_SNR$ (11)
    - If $SNR < Max_SNR$, then:

$$ISNR_Prev = INT\left[\frac{SNR - Min_SNR}{Step_SNR}\right] \tag{12}$$

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Else ($SNR \geq Max_SNR$), then:

$$SNR = Max_SNR \tag{13}$$

$$ISNR_Prev = INT\left[\frac{SNR - Min_SNR}{Step_SNR}\right] - 1 \tag{14}$$

* $ISNR_Next = ISNR_Prev + 1$ (15)

– The corresponding signal to noise ratios are computed by:

* $SNR_Prev = Min_SNR + ISNR_Prev * Step_SNR$ (16)
* $SNR_Next = SNR_Prev + Step_SNR$ (17)

- The modeled instrumental correction is then computed by:

$$\begin{aligned} Cor_Mod = & \frac{(SWH_Next - SWH) * (SNR_Next - SNR)}{Step_SWH * Step_SNR} * Tab(ISWH_Prev, ISNR_Prev) \\ & + \frac{(SWH_Next - SWH) * (SNR - SNR_Prev)}{Step_SWH * Step_SNR} * Tab(ISWH_Prev, ISNR_Next) \\ & + \frac{(SWH - SWH_Prev) * (SNR_Next - SNR)}{Step_SWH * Step_SNR} * Tab(ISWH_Next, ISNR_Prev) \\ & + \frac{(SWH - SWH_Prev) * (SNR - SNR_Prev)}{Step_SWH * Step_SNR} * Tab(ISWH_Next, ISNR_Next) \end{aligned} \tag{18}$$

COMMENTS

The modeled correction of instrumental errors on the altimetric parameters are made using correction tables depending on significant waveheight and signal to noise ratio. Bilinear interpolation is used.

REFERENCES

None

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_COR_05 - Calculation of a path delay value

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J. STUM CLS	
Checked by:	For the JASON-1 SWT L. EYMARD For the JASON-1 Project P. VINCENT CNES S. DESAI JPL For the F-PAC / ENVISAT Project J. BENVENISTE ESA P. VINCENT CNES	
Approved by:	P. VINCENT CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To compute a path delay value from logarithmic function of three input brightness temperatures. The coefficients of the function are derived by linear interpolation in wind speed of input tabulated coefficients.

ALGORITHM SPECIFICATION

Input data

- First value of tabulated coefficient B0 : Val1_TAB_B0 (m)
- First value of tabulated coefficient B1 : Val1_TAB_B1 (m/Log(K))
- First value of tabulated coefficient B2 : Val1_TAB_B2 (m/Log(K))
- First value of tabulated coefficient B3 : Val1_TAB_B3 (m/Log(K))
- Second value of tabulated coefficient B0 : Val2_TAB_B0 (m)
- Second value of tabulated coefficient B1 : Val2_TAB_B1 (m/Log(K))
- Second value of tabulated coefficient B2 : Val2_TAB_B2 (m/Log(K))
- Second value of tabulated coefficient B3 : Val2_TAB_B3 (m/Log(K))
- Weight to be applied to the first values : W1 (/)
- Weight to be applied to the second values : W2 (/)
- The three brightness temperatures : TB1, TB2, TB3 (K)

Output data

- The path delay value : Path_Delay (m)
- Execution status

Processing

- Computing by linear interpolation in wind speed the retrieval coefficient B0, using mechanism "GEN_MEC_INT_02 - Linear interpolation",

  the input parameters of which are:

  – X1 = Val1_TAB_B0
  – X2 = Val2_TAB_B0
  – W1 = w1
  – W2 = w2

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

and the output parameter of which is:

- B0

- Computing by linear interpolation in wind speed the retrieval coefficient B1, using mechanism "GEN_MEC_INT_02 - Linear interpolation",

  the input parameters of which are:

  - X1 = Val1_TAB_B1
  - X2 = Val2_TAB_B1
  - W1 = w1
  - W2 = w2

  and the output parameter of which is:

  - B1

- Computing by linear interpolation in wind speed the retrieval coefficient B2, using mechanism "GEN_MEC_INT_02 - Linear interpolation",

  the input parameters of which are:

  - X1 = Val1_TAB_B2
  - X2 = Val2_TAB_B2
  - W1 = w1
  - W2 = w2

  and the output parameter of which is:

  - B2

- Computing by linear interpolation in wind speed the retrieval coefficient B3, using mechanism "GEN_MEC_INT_02 - Linear interpolation",

  the input parameters of which are:

  - X1 = Val1_TAB_B3
  - X2 = Val2_TAB_B3
  - W1 = w1
  - W2 = w2

  and the output parameter of which is:

  - B3

- Computing the path delay value:

  - $\text{Path_Delay} = B0 + B1 * \text{Log}_e(280 - TB1) + B2 * \text{Log}_e(280 - TB2) + B3 * \text{Log}_e(280 - TB3)$ (1)

COMMENTS

This mechanism is not called if one of the brightness temperatures is > 279.9 K.

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

REFERENCES

None

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_CON_01 - Derivation of the significant waveheight from the composite Sigma

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J.P. DUMONT CLS	
Checked by:	For the JASON-1 SWT D.HANCOCK E. RODRIGUEZ M. SROKOSZ For the JASON-1 Project P. VINCENT CNES S. DESAI JPL For the F-PAC / ENVISAT Project J. BENVENISTE ESA P. VINCENT CNES	
Approved by:	P. VINCENT CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To derive the significant waveheight from the composite Sigma.

ALGORITHM SPECIFICATION

Input data

- Composite Sigma : SigmaC (s)
- Composite Sigma validity flag [(1)] : SigmaC_Val_Flag (/)
- PTR width:
  - Sampling interval of the analysis window : FFT_Step (s)
  - Ratio PTR width / FFT step : Ratio_PTR_FFT (/)
- Light velocity : Light_Vel (m/s)

Output data

- Significant waveheight : SWH (m)
- Significant waveheight validity flag [(1)] : SWH_Val_Flag (/)
- Execution status

Processing

- If the validity flag for the retracking-derived parameter (SigmaC_Val_Flag) is set to "valid", then:

$$SigmaP = Ratio_PTR_FFT * FFT_Step \quad (1)$$

  - If $SigmaC^2 > SigmaP^2$, then:

$$SWH = 2 * Light_Vel * \sqrt{SigmaC^2 - SigmaP^2} \quad (2)$$

  - Else ($SigmaC^2 \leq SigmaP^2$), then:

$$SWH = 0 \quad (3)$$

  - The corresponding validity flag (SWH_Val_Flag) is set to valid
- Else (SigmaC_Val_Flag set to "invalid"):
  - SWH is set to a default value
  - The corresponding validity flag (SWH_Val_Flag) is set to "invalid"

[(1)] 2 states: "valid" or "invalid"

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

COMMENTS

None

REFERENCES

None

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_CON_02 - Derivation of the square of the off-nadir angle from the slope of the trailing edge

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J.P. DUMONT CLS	
Checked by:	For the JASON-1 SWT G. HAYNE E. RODRIGUEZ For the JASON-1 Project P. VINCENT CNES S. DESAI JPL For the F-PAC / ENVISAT Project J. BENVENISTE ESA P. VINCENT CNES	
Approved by:	P. VINCENT CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To derive the square of the off-nadir angle from the slope of the trailing edge.

ALGORITHM SPECIFICATION

Input data

- Gamma instrumental parameter : Gamma (/)
- Alpha instrumental parameter : Alpha (s^{-1})
- Slope of the logarithm of the trailing edge : Slope (s^{-1})
- Flags [(1)]
  - Retracking execution flag : Retrack_Flag
  - Slope computation flag : Slope_Flag

Output data

- Square of the off-nadir angle : Sq_Off_Nad ($degree^2$)
- Validity flag for the square of the off-nadir angle [(1)] : Sq_Off_Nad_Val_Flag (/)
- Execution status

Processing

- If the retracking execution flag (Retrack_Flag) and the slope computation flag (Slope_Flag) are set to "valid", then:
  - The square of the off-nadir angle is computed by:

$$Sq_Off_Nad = \left(\frac{180}{\pi}\right)^2 * \frac{1+\frac{Slope}{Alpha}}{2*\left(1+\frac{2}{Gamma}\right)} \qquad (1)$$

  - The corresponding validity flag (Sq_Off_Nad_Val_Flag) is set to "valid"
- Else:
  - Sq_Off_Nad_Val_Flag is set to "invalid"
  - Sq_Off_Nad is set to a default value.

---

(1) 2 states: "valid" or "invalid"

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

COMMENTS

"Gamma" and "Alpha" parameters (γ and α) are defined by:

$$\gamma = \frac{2}{\mathrm{Log}_e(2)}.\sin^2\left(\frac{\theta_0}{2}\right) \quad \text{and} \quad \alpha = \frac{4c}{\gamma h\left(1+\frac{h}{R_e}\right)}$$

where θ_0 is the antenna beamwidth, c is the light velocity, h is the mean satellite altitude and R_e is the earth radius.

REFERENCES

None

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_CON_03 - Conversion of a position vector from geodetic to Cartesian co-ordinates

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J.P. DUMONT CLS	
Checked by:	For the JASON-1 SWT F. LEMOINE C.K. SHUM For the JASON-1 Project P. VINCENT CNES S. DESAI JPL For the F-PAC / ENVISAT Project J. BENVENISTE ESA P. VINCENT CNES	
Approved by:	P. VINCENT CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To convert a position vector from geodetic co-ordinates (elevation above the reference ellipsoid, geodetic latitude and longitude) to a vector in Cartesian co-ordinates (X-component, Y-component and Z-component).

ALGORITHM SPECIFICATION

Input data

- Geodetic components of the position vector:
  - Elevation above the reference ellipsoid : Elev (m)
  - Geodetic latitude : Lat (degree)
  - Longitude : Lon (degree [0,360[)
- Characteristics of the reference ellipsoid:
  - Semi major axis : SM_Axis (m)
  - Square of the eccentricity : Ecc2 (/)

Output data

- Cartesian components of the position vector:
  - X-component : X (m)
  - Y-component : Y (m)
  - Z-component : Z (m)
- Execution status

Processing

- The following intermediate parameter is computed:

$$\text{Temp} = \frac{\text{SM_Axis}}{\sqrt{1 - \text{Ecc2} * \sin^2\left(\frac{\pi}{180} * \text{Lat}\right)}} \tag{1}$$

- The Cartesian components (X, Y and Z) of the position vector are then derived by:

$$X = (\text{Temp} + \text{Elev}) * \cos\left(\frac{\pi}{180} * \text{Lat}\right) * \cos\left(\frac{\pi}{180} * \text{Lon}\right) \tag{2}$$

$$Y = (\text{Temp} + \text{Elev}) * \cos\left(\frac{\pi}{180} * \text{Lat}\right) * \sin\left(\frac{\pi}{180} * \text{Lon}\right) \tag{3}$$

$$Z = [\text{Temp} * (1 - \text{Ecc2}) + \text{Elev}] * \sin\left(\frac{\pi}{180} * \text{Lat}\right) \tag{4}$$

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

COMMENTS

None

REFERENCES

- Nouel, F., Les Repères de l'Espace et du Temps, Le Mouvement du Véhicule Spatial en orbite, Cours de Technologie Spatiale, CNES, 1980.

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_CON_04 - Time conversions

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J.P. DUMONT CLS J. STUM CLS	
Checked by:	For the JASON-1 SWT P. VINCENT CNES For the JASON-1 Project P. VINCENT CNES S. DESAI JPL For the F-PAC / ENVISAT Project J. BENVENISTE ESA P. VINCENT CNES	
Approved by:	P. VINCENT CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To perform time conversions.

ALGORITHM SPECIFICATION

This function will be implemented as a set of functions, each of them being devoted to one type of conversion.

Function "From DaySecMic1950 To Sec1950"

This function converts a TAI time elapsed since 01/01/50 at 0 h from days-seconds-microseconds (integers) to seconds (real).

Input data

- Time elapsed since 01/01/50 at 0 h:
  - Days : Day_In (integer)
  - Seconds in the day : Sec_In (integer)
  - Microseconds in the second : Mic_In (integer)

Output data

- Seconds elapsed since 01/01/50 at 0 h. : Sec_Out (real)

Processing

$$Sec_Out = Day_In * 86400 + Sec_In + \frac{Mic_In}{10^6} \quad (1)$$

Function "From SecUTC1950 To Day1900"

This function converts an UTC time from seconds elapsed since 01/01/50 at 0 h. (real) to days elapsed since 01/01/1900 at 0 h. (real).

Input data

- Seconds elapsed since 01/01/50 at 0 h. (UTC) : Sec_In (real)

Output data

- Days elapsed since 01/01/1900 at 0 h. : Day_Out (real)

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Processing

$$Day_Out = \frac{Sec_In}{86400} + 18262 \quad (1)$$

Function "From SecUTC1950 To Expanded1"

This function converts an UTC time elapsed since 01/01/50 at 0 h. from seconds (real) to year (integer), day of the year (integer between 1 and 366), month number (integer between 1 and 12), day of the month (integer between 1 and 31), seconds of the day (real) and decimal hour (real).

Input data

- Seconds elapsed since 01/01/50 at 0 h. (UTC) : Sec_1950 (real)

Output data

- Year : Year_Out (integer)
- Day of the year : Day_Out (integer)
- Month number : MonthNb_Out (integer)
- Day of the month : Day_Month_Out (integer)
- Seconds of the day : Sec_Out (real)
- Decimal hour : Hour_Out (real)

Processing

- The number of days elapsed since 01/01/1950 is computed by :

$$Day_1950 = FLOOR\left(\frac{Sec_1950}{86400}\right) \quad (1)$$

- The number of seconds in the day is computed by :

$$Sec_Out = Sec_1950 - Day_1950 * 86400 \quad (2)$$

- The decimal hour is computed by :

$$Hour_Out = \frac{Sec_Out}{3600} \quad (3)$$

- The number of years elapsed since 01/01/1950 is computed by :

$$Year_1950 = FLOOR\left(\frac{Day_1950}{365.25}\right) \quad (4)$$

- The year is computed by :

$$Year_Out = 1950 + Year_1950 \quad (5)$$

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- The number of leap years elapsed since 01/01/1950 is computed by :

$$\text{Year_1950_Leap} = \text{INT}\left(\frac{\text{Year_Out} - 1901}{4}\right) - 12 \quad (6)$$

- The number of days in the current year is computed by :

$$\text{Day_Out} = \text{Day_1950} - \text{Year_1950} * 365 - \text{Year_1950_Leap} + 1 \quad (7)$$

If $\text{Day_Out} \leq 0$, then $\text{Day_Out} = \text{Day_Out} + 365$

- Test on the current year (MOD is the modulo function) :

If $\text{MOD(Year_Out,4)} = 0$, then $j = 2$, else $j = 1$ (8)

- The month of the year is computed by :

$$\text{MonthNb_Out} = i \quad (9)$$

where i is determined as follows :

$$Q_j(i-1) < \text{Day_Out} \leq Q_j(i)$$

with :

- $Q_1(0) = 0$ $Q_2(0) = 0$
- $Q_1(1) = 31$ $Q_2(1) = 31$
- $Q_1(2) = 59$ $Q_2(2) = 60$
- $Q_1(3) = 90$ $Q_2(3) = 91$
- $Q_1(4) = 120$ $Q_2(4) = 121$
- $Q_1(5) = 151$ $Q_2(5) = 152$
- $Q_1(6) = 181$ $Q_2(6) = 182$
- $Q_1(7) = 212$ $Q_2(7) = 213$
- $Q_1(8) = 243$ $Q_2(8) = 244$
- $Q_1(9) = 273$ $Q_2(9) = 274$
- $Q_1(10) = 304$ $Q_2(10) = 305$
- $Q_1(11) = 334$ $Q_2(11) = 335$
- $Q_1(12) = 365$ $Q_2(12) = 366$

- The day of the month is computed by :

$$\text{Day_Month} = \text{Day_Out} - Q_j(i-1)$$

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Function "From Expanded2 to SecUTC1950"

This function converts a time from day, month, year, hour, minutes, seconds and microseconds to UTC seconds elapsed since 01/01/50 at 0 h.

Input data

- Day of the month : Day_In (integer)
- Month id. : Month_In (caracters)
- Year : Year_In (integer)
- Hour : Hour_In (integer)
- Minutes : Min_In (integer)
- Seconds : Sec_In (integer)
- Microseconds : Micro_In (integer)

Output data

- Seconds elapsed since 01/01/50 at 0 h. (UTC) : Sec_Out (real)

Processing

- The number of leap years elapsed since 01/01/1950 is computed by :

$$\text{Year_1950_Leap} = \text{INT}\left(\frac{\text{Year_In} - 1901}{4}\right) - 12 \qquad (1)$$

- Test on the current year (MOD is the modulo function) :

$$\text{If MOD(Year_In,4)} = 0\text{, then } j = 2\text{, else } j = 1 \qquad (2)$$

- Month_In is converted in the number of the month in the year (1 for JAN, 2 for FEB, ..., 12 for DEC)

- The number of seconds elapsed since 01/01/1950 is computed by :

$$\begin{aligned}\text{Sec_Out} = &\left[(\text{Year_In} - 1950) * 365 + \text{Year_1950_Leap} + Q_j(\text{Month_In} - 1) + \text{Day_In} - 1\right] * 86400 \\ &+ \text{Hour_In} * 3600 + \text{Min_In} * 60 + \text{Sec_In} + \text{Micro_In} * 10^{-6}\end{aligned} \qquad (3)$$

with :

- $Q_1(0) = 0$ $Q_2(0) = 0$
- $Q_1(1) = 31$ $Q_2(1) = 31$
- $Q_1(2) = 59$ $Q_2(2) = 60$
- $Q_1(3) = 90$ $Q_2(3) = 91$
- $Q_1(4) = 120$ $Q_2(4) = 121$

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

	$Q_1(5)$ = 151	$Q_2(5)$	= 152
–	$Q_1(6)$ = 181	$Q_2(6)$	= 182
–	$Q_1(7)$ = 212	$Q_2(7)$	= 213
–	$Q_1(8)$ = 243	$Q_2(8)$	= 244
–	$Q_1(9)$ = 273	$Q_2(9)$	= 274
–	$Q_1(10)$ = 304	$Q_2(10)$	= 305
–	$Q_1(11)$ = 334	$Q_2(11)$	= 335
–	$Q_1(12)$ = 365	$Q_2(12)$	= 366

Function "From Expanded1 to SecUTC1950"

This function converts a time from day, month, year, hour, minutes, seconds and microseconds to UTC seconds elapsed since 01/01/50 at 0 h.

Input data

- Day of the month : Day_In (integer)
- Month : Month_In (integer)
- Year : Year_In (integer)
- Hour : Hour_In (integer)
- Minutes : Min_In (integer)
- Seconds : Sec_In (integer)
- Microseconds : Micro_In (integer)

Output data

- Seconds elapsed since 01/01/50 at 0 h. (UTC) : Sec_Out (real)

Processing

- The number of leap years elapsed since 01/01/1950 is computed by :

$$\text{Year_1950_Leap} = \text{INT}\left(\frac{\text{Year_In} - 1901}{4}\right) - 12 \qquad (1)$$

- Test on the current year (MOD is the modulo function) :

$$\text{If MOD(Year_In,4)} = 0\text{, then } j = 2\text{, else } j = 1 \qquad (2)$$

- The number of seconds elapsed since 01/01/1950 is computed by :

$$\begin{aligned}\text{Sec_Out} = &\left[(\text{Year_In} - 1950) * 365 + \text{Year_1950_Leap} + Q_j(\text{Month_In} - 1) + \text{Day_In} - 1\right] * 86400 \\ &+ \text{Hour_In} * 3600 + \text{Min_In} * 60 + \text{Sec_In} + \text{Micro_In} * 10^{-6}\end{aligned} \qquad (3)$$

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

with :

- $Q_1(0) = 0$ $Q_2(0) = 0$
- $Q_1(1) = 31$ $Q_2(1) = 31$
- $Q_1(2) = 59$ $Q_2(2) = 60$
- $Q_1(3) = 90$ $Q_2(3) = 91$
- $Q_1(4) = 120$ $Q_2(4) = 121$
- $Q_1(5) = 151$ $Q_2(5) = 152$
- $Q_1(6) = 181$ $Q_2(6) = 182$
- $Q_1(7) = 212$ $Q_2(7) = 213$
- $Q_1(8) = 243$ $Q_2(8) = 244$
- $Q_1(9) = 273$ $Q_2(9) = 274$
- $Q_1(10) = 304$ $Q_2(10) = 305$
- $Q_1(11) = 334$ $Q_2(11) = 335$
- $Q_1(12) = 365$ $Q_2(12) = 366$

COMMENTS

Algorithms that require leap year information using a "MOD(Year,4) = 0" check will fail in the year 2100. This is because leap years fall on all years divisible by 4, unless the year is divisible by 100 then it is not a leap year, and unless the year is divisible by 400 then it is a leap year.

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_CON_05 - Conversion between geographic and geomagnetic latitude

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	S. LABROUE	CLS	
Checked by:	N. PICOT	CNES	
Approved by:	P. VINCENT	CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To convert geographic latitude into geomagnetic latitude.

ALGORITHM SPECIFICATION

Input data

- Latitude to be converted (geographic) : Lat (degrees)
- Longitude of the point : Lon (degrees)
- Table of values of magnetic dipole function of geographic latitude and longitude : Tab_dip[0:Nb_lon-1][0:Nb_lat-1] (radians)
- Characteristics of the grid :
  - Number of grid points in the longitude axis : Nb_lon (/)
  - Number of grid points in the latitude axis : Nb_lat (/)
  - Grid step in the longitude axis : Lon_step (degrees)
  - Grid step in the latitude axis : Lat_step (degrees)
  - First tabulated latitude value : Lon_first (degrees)
  - First tabulated longitude value : Lat_first (degrees)

Output data

- Converted latitude : Lat_out (radians)
- Output flag : Flag_grid [1]
- Execution status

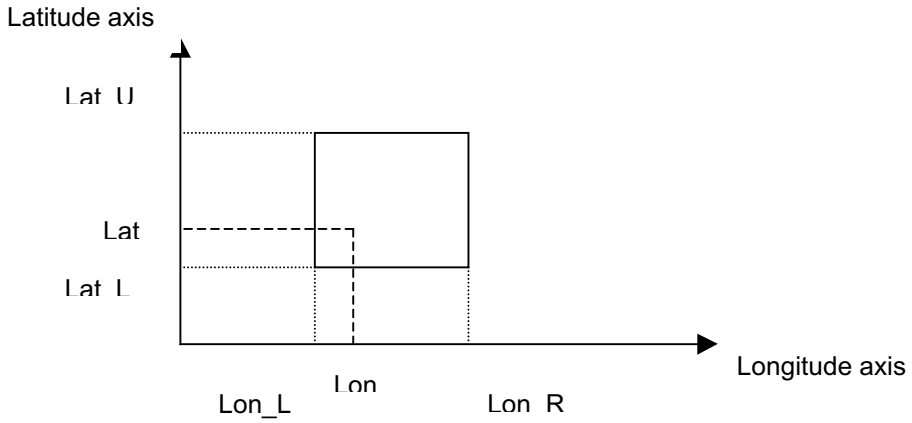
Processing

To compute geomagnetic latitude (equal to magnetic dipole) at point (Lat, Lon), bilinear interpolation of magnetic dipole, function of geographic latitude and longitude, is performed.

Using the four points surrounding the point (Lat,Lon) in the two dimension grid in latitude, longitude, the four values of magnetic dipole are determined from the table of values of magnetic dipole. The interpolated value of magnetic dipole at point (Lat, Lon) is then computed.

[1] 2 states: "valid" or "invalid"

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.



- Normalization of the longitude (Lon)

  Longitude of the point is translated of 360 such that $0 \leq Lon < 360$

- To select the grid points surrounding the point (Lon, Lat) :

  Using mechanism "GEN_MEC_GRI_01 - Cell identification" with the following inputs :

  – X co-ordinate of the point : Lon
  – Y co-ordinate of the point : Lat
  – Table step (or grid step) in X : Lon_step
  – Table step (or grid step) in Y : Lat_step
  – Value of X corresponding to the first table point (or grid point) : Lon_first
  – Value of Y corresponding to the first table point (or grid point) : Lat_first
  – Number of points in X of the table (or the grid) : Nb_lon
  – Number of points in Y of the table (or the grid) : Nb_lat
  – Cycling value for the X variable : 0
  – Cycling value for the Y variable : 0
  – Truncation flag for the X variable : 1
  – Truncation flag for the Y variable : 1

  The following outputs are computed :

  – Indexes of the four table points (or grid points) surrounding the (X,Y) point:
    ∗ Left index in X : Lon_left (/)
    ∗ Right index in X : Lon_right (/)
    ∗ Bottom index in Y : Lat_bottom (/)
    ∗ Top index in Y : Lat_top (/)

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- Weight of these four points:
  - Weight of the lower left corner : W_LL (/)
  - Weight of the lower right corner : W_LR (/)
  - Weight of the upper left corner : W_UL (/)
  - Weight of the upper right corner : W_UR (/)

- To determine the four values of magnetic dipole at the four points, using the table of values of magnetic dipole :

  Dip_LL = Tab_dip(Lon_left, Lat_bottom)

  Dip_LR = Tab_dip(Lon_right, Lat_bottom)

  Dip_UL = Tab_dip(Lon_left, Lat_top)

  Dip_UR = Tab_dip(Lon_right, Lat_top)

- Geomagnetic latitude equal to magnetic dipole (Lat_out) is computed using mechanism “GEN_MEC_INT_03 - Bilinear interpolation” with the following inputs :
  - Values of the four corners of the cell:
    - Value of the lower left corner : Dip_LL
    - Value of the lower right corner : Dip_LR
    - Value of the upper left corner : Dip_UL
    - Value of the upper right corner : Dip_UR
  - Weights of the four corners of the cell:
    - Weight of the lower left corner : W_LL (/)
    - Weight of the lower right corner : W_LR (/)
    - Weight of the upper left corner : W_UL (/)
    - Weight of the upper right corner: : W_UR (/)
  - Default value of a corner : unused value
- If the interpolated value Lat_out has not a default value then the output flag Flag_grid is set to "valid"

COMMENTS

None

REFERENCES

None

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_CON_06 - Conversion of a position vector from Cartesian to geodetic co-ordinates

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J.P. DUMONT CLS	
Checked by:	For the JASON-1 SWT P. VINCENT CNES For the JASON-1 Project P. VINCENT CNES S. DESAI JPL For the F-PAC / ENVISAT Project J. BENVENISTE ESA P. VINCENT CNES	
Approved by:	P. VINCENT CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To convert a position vector from Cartesian co-ordinates (X-component, Y-component and Z-component) to a vector in geodetic co-ordinates (elevation above the reference ellipsoid, geodetic latitude and longitude).

ALGORITHM SPECIFICATION

Input data

- Position of the satellite:
  - X co-ordinate : PSx (m)
  - Y co-ordinate : PSy (m)
  - Z co-ordinate : PSz (m)
- Characteristics of the reference ellipsoid:
  - Semi major axis : SM_Axis (m)
  - Flattening : Flattening (/)
- Thresholds for the iterative process:
  - Desired accuracy for the orbit altitude : Acc_Orb_Alt (m)
  - Desired accuracy for the latitude : Acc_Lat (degree)

Output data

- Orbit altitude : Orb_Alt (m)
- Latitude : Lat (degree)
- Longitude : Lon (degree [0, 360[)
- Execution status

Processing

- First, the square of the eccentricity (Ecc2) is computed:

$$\text{Ecc2} = \text{Flattening} * (2 - \text{Flattening}) \quad (1)$$

- The longitude (Lon $\in$ [0, 360[degrees) is computed as follows where the result of the Arctg function is defined in the interval]-π/2, π/2[: [1]
  - Lon = 0 if PSy = 0 and PSx > 0 (2)

[1] The case PSx = PSy = 0 should not occur

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

– $\text{Lon} = 90 - \frac{180}{\pi} * \text{Arctg}\left(\frac{\text{PSx}}{\text{PSy}}\right)$ if PSy > 0 (3)

– $\text{Lon} = 180$ if PSy = 0 and PSx < 0 (4)

– $\text{Lon} = 270 - \frac{180}{\pi} * \text{Arctg}\left(\frac{\text{PSx}}{\text{PSy}}\right)$ if PSy < 0 (5)

- Then, the latitude (Lat $\in$ [-90, 90] degrees) and the orbit altitude (Orb_Alt) are computed using the following iterative process:
  - Initialization of the iterative process:
    - The geocentric distance of the satellite (GCdistS) is computed by:

$$\text{GCdistS} = \sqrt{\text{PSx}^2 + \text{PSy}^2 + \text{PSz}^2} \quad (6)$$

    - The declination (Declin $\in$ [-90, 90] degrees) of the satellite position is computed by:

$$\text{Declin} = \frac{180}{\pi} * \text{Arctg}\left(\frac{\text{PSz}}{\sqrt{\text{PSx}^2 + \text{PSy}^2}}\right) \quad (7)$$

    - The geocentric latitude (GClat $\in$ [-90, 90] degrees), the geodetic latitude (GDlat $\in$ [-90, 90] degrees) and the geodetic altitude of the satellite (GDaltS) are initialized by:

$$\text{GClat}_0 = \text{Declin} \quad (8)$$

$$\text{GDlat}_0 = \frac{180}{\pi} * \text{Arctg}\left[\frac{\text{tg}\left(\frac{\pi}{180} * \text{GClat}_0\right)}{(1 - \text{Flattening})^2}\right] \quad (9)$$

$$\text{GDaltS}_0 = \text{GCdistS} \quad (10)$$

  - The geodetic altitude of the satellite, the geocentric distance of the nadir point (GCdistN), and the geocentric and geodetic latitudes of the nadir point are then computed iteratively (loop index i starting from 1) by the following set of equations (11) to (19):
    - Computation of the geocentric distance of the nadir point:

$$\text{GCdistN}_i = \frac{\text{SM_Axis} * (1 - \text{Flattening})}{\sqrt{1 - \text{Ecc2} * \cos^2\left(\frac{\pi}{180} * \text{GClat}_{i-1}\right)}} \quad (11)$$

    - Computation of the topocentric aspect angle (TAangle):

$$\text{TAangle}_i = \text{GDlat}_{i-1} - \text{GClat}_{i-1} \quad (12)$$

    - Computation of the geodetic altitude of the satellite:

$$\text{GDaltS}_i = \sqrt{\text{GCdistS}^2 - \text{GCdistN}_i^2 * \sin^2\left(\frac{\pi}{180} * \text{TAangle}_i\right)} - \text{GCdistN}_i * \cos\left(\frac{\pi}{180} * \text{TAangle}_i\right) \quad (13)$$

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

* Computation of the geocentric aspect angle:

$$GAangle_i = \frac{180}{\pi} * \arcsin\left[\frac{GDaltS_i}{GCdistS} * \sin\left(\frac{\pi}{180} * TAangle_i\right)\right] \quad (14)$$

* Computation of the geocentric latitude:

$$GClat_i = Declin - GAangle_i \quad (15)$$

* Computation of the geodetic latitude:

$$GDlat_i = \frac{180}{\pi} * Arctg\left[\frac{tg\left(\frac{\pi}{180} * GClat_i\right)}{(1 - Flattening)^2}\right] \quad (16)$$

* Test on the precision of the estimates:
  - ◊ If the following condition is satisfied:

$$\left|GDaltS_i - GDaltS_{i-1}\right| \le Acc_Orb_Alt \quad \text{and} \quad \left|GDlat_i - GDlat_{i-1}\right| \le Acc_Lat \quad (17)$$

  - ◊ Then:

$$Orb_Alt = GDaltS_i \quad (18)$$

$$Lat = GDlat_i \quad (19)$$

  - ◊ Else return to equation (11).

REFERENCES

- Nouel, F., Les Repères de l'Espace et du Temps, Le Mouvement du Véhicule Spatial en orbite, Cours de Technologie Spatiale, CNES, 1980.
- Klinkrad, H., ERS-1, Algorithms for orbit prediction and for the determination of related static and dynamic altitude and groundtrace quantities, ESA, ER-RP-ESA-SY-0001, 1985.

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_MOD_01 - Ocean echo model and partial derivatives: Initialization (1)

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J.P. DUMONT CLS	
Checked by:	For the JASON-1 SWT G. HAYNE For the JASON-1 Project P. VINCENT CNES S. DESAI JPL For the F-PAC / ENVISAT Project J. BENVENISTE ESA P. VINCENT CNES	
Approved by:	P. VINCENT CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To initialize the computation of the ocean echo model and of its partial derivatives (first step).

ALGORITHM SPECIFICATION

Input data

- Altimetric estimates:
  - Composite sigma : SigmaC (s)
  - Amplitude : Ampl (FFT power unit)
- A and B parameters:
  - A parameter : A (/)
  - B parameter : B (/)
- Instrumental parameters:
  - Alpha parameter : Alpha (s^{-1})
  - PTR width : PTR_Width (s)
- Skewness coefficient : Skew (/)

Output data

- M, L, C, D, C12 and K1 parameters:
  - M parameter : M (s^{-1})
  - L parameter : L (/)
  - C parameter : C (s^{-1})
  - D parameter : D (/)
  - C12 parameter : C12 (/)
  - K1 parameter : K1 (FFT power unit)
- Execution status

Processing

- If $\mathrm{SigmaC}^2 > \mathrm{PTR_Width}^2$, then:

$$\mathrm{SigmaS} = \sqrt{\mathrm{SigmaC}^2 - \mathrm{PTR_Width}^2} \tag{1}$$

$$\mathrm{M} = \frac{\mathrm{SigmaC}^2 - \mathrm{SigmaS}^2}{\mathrm{SigmaS}^2 * \mathrm{SigmaC}} \tag{2}$$

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Else:

$$SigmaS = 0 \tag{3}$$

$$M = 0 \tag{4}$$

- $$L = Skew * \left(\frac{SigmaS}{SigmaC}\right)^3 \tag{5}$$
- $$C = B * Alpha \tag{6}$$
- $$D = C * SigmaC \tag{7}$$
- $$C12 = 1 + \frac{L}{6} * D^3 \tag{8}$$
- $$K1 = A * Ampl \tag{9}$$

COMMENTS

A global mathematical for the computation of the ocean echo model and partial derivatives is given in section "Comments" of "GEN_MEC_MOD_03 - Ocean echo model and partial derivatives: Computation".

REFERENCES

None

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_MOD_02 - Ocean echo model and partial derivatives: Initialization (2)

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J.P. DUMONT CLS	
Checked by:	For the JASON-1 SWT G. HAYNE For the JASON-1 Project P. VINCENT CNES S. DESAI JPL For the F-PAC / ENVISAT Project J. BENVENISTE ESA P. VINCENT CNES	
Approved by:	P. VINCENT CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To initialize the computation of the ocean echo model and of its partial derivatives (second step).

ALGORITHM SPECIFICATION

Input data

- Abscissa of the sample to be processed : Abs_Sample (/)
- Abscissa of the reference sample for tracking : Abs_Ref (/)
- Sampling interval of the analysis window : FFT_Step (s)
- Altimetric estimates:
  - Epoch : Epoch (s)
  - Composite sigma : SigmaC (s)
- C and D parameters:
  - C parameter : C (s^{-1})
  - D parameter : D (/)

Output data

- U, Y and V parameters:
  - U parameter : U (/)
  - Y parameter : Y (/)
  - V parameter : V (s^{-1})
- Execution status

Processing

- $T = [Abs_Sample - Abs_Ref] * FFT_Step$ (1)
- $U = \frac{T - Epoch}{SigmaC} - D$ (2)
- $Y = \frac{U}{\sqrt{2}}$ (3)
- $V = \frac{T - Epoch}{SigmaC^2} + C$ (4)

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

COMMENTS

A global mathematical for the computation of the ocean echo model and partial derivatives is given in section "Comments" of "GEN_MEC_MOD_03 - Ocean echo model and partial derivatives: Computation".

REFERENCES

None

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_MOD_03 - Ocean echo model and partial derivatives: Computation

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J.P. DUMONT CLS	
Checked by:	For the JASON-1 SWT G. HAYNE For the JASON-1 Project P. VINCENT CNES S. DESAI JPL For the F-PAC / ENVISAT Project J. BENVENISTE ESA P. VINCENT CNES	
Approved by:	P. VINCENT CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	Creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To compute a sample of:

- The ocean echo model
- The partial derivatives of the ocean echo model with respect to the epoch
- The partial derivatives of the ocean echo model with respect to the composite sigma
- The partial derivatives of the ocean echo model with respect to the amplitude

ALGORITHM SPECIFICATION

Input data

- Altimetric estimates:
  - Composite sigma : SigmaC (s)
  - Thermal noise level : TN (FFT power unit)
- A parameter : A (/)
- M, L, C, D, C12 and K1 parameters:
  - M parameter : M (s^{-1})
  - L parameter : L (/)
  - C parameter : C (s^{-1})
  - D parameter : D (/)
  - C12 parameter : C12 (/)
  - K1 parameter : K1 (FFT power unit)
- U, Y and V parameters:
  - U parameter : U (/)
  - Y parameter : Y (/)
  - V parameter : V (s^{-1})
- Limit argument for the erf function (absolute value) : Arg_Limit (/)
- Processing option [1] : Option

[1] 3 states: "Model", "Derivatives" or "Model and Derivatives"

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Output data

- Sample of:
  - The echo model : Mod (FFT power unit)
  - The partial derivative of the echo model w.r.t. epoch : D_Mod_Epoch (FFT power unit/s)
  - The partial derivative of the echo model w.r.t. composite sigma : D_Mod_SigmaC (FFT power unit/s)
  - The partial derivative of the echo model w.r.t. amplitude : D_Mod_Ampl (/)
- Execution status

Processing

- **If Y < -Arg_Limit, then:**
  - If Option is set to "Model" or "Model and Derivatives", then:

$$\boxed{Mod = TN} \tag{1}$$

  - If Option is set to "Derivatives" or "Model and Derivatives", then:

$$\boxed{D_Mod_Epoch = 0} \tag{2}$$

$$\boxed{D_Mod_SigmaC = 0} \tag{3}$$

$$\boxed{D_Mod_Ampl = 0} \tag{4}$$

- **If Y > Arg_Limit, then:**

$$K2 = \exp\left[-D * \left(U + \frac{D}{2}\right)\right] \tag{5}$$

$$K = K1 * K2 \tag{6}$$

  - If Option is set to "Model" or "Model and Derivatives", then:

$$\boxed{Mod = K * C12 + TN} \tag{7}$$

  - If Option is set to "Derivatives" or "Model and Derivatives", then:

$$\boxed{D_Mod_Epoch = \frac{K}{SigmaC} * D * C12} \tag{8}$$

$$\boxed{D_Mod_SigmaC = K * \left[C^2 * SigmaC * C12 + \frac{L * D^2}{2} * (M * D + C)\right]} \tag{9}$$

$$\boxed{D_Mod_Ampl = A * K2 * C12} \tag{10}$$

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- **If -Arg_Limit ≤ Y ≤ Arg_Limit, then:**

$$C11 = \frac{1}{2} * [1 + \mathrm{erf}(Y)] \tag{11}$$

$$C1 = C11 * C12 \tag{12}$$

$$C21 = \frac{1}{\sqrt{2\pi}} * \exp\left[-Y^2\right] \tag{13}$$

$$C22 = -\frac{L}{6} * (U^2 + 3 * D * U + 3 * D^2 - 1) \tag{14}$$

$$C2 = C21 * C22 \tag{15}$$

$$C3 = C1 + C2 \tag{16}$$

K2 is computed according to (5)

K is computed according to (6)

– If Option is set to "Model" or "Model and Derivatives", then:

$$\boxed{Mod = K * C3 + TN} \tag{17}$$

– If Option is set to "Derivatives" or "Model and Derivatives", then:

$$X = 2 * U + 3 * D \tag{18}$$

$$\boxed{D_Mod_Epoch = \frac{K}{SigmaC} * \left[D * C3 + C21 * \left(-C12 + \frac{L}{6} * X + C22 * U\right)\right]} \tag{19}$$

$$\boxed{D_Mod_SigmaC = K * \left\{\begin{array}{l} C^2 * SigmaC * C3 + C11 * \frac{L * D^2}{2} * (M * D + C) \\ -C21 * \left[C12 * V - C22 * (3 * M + U * V) + \frac{L}{6} * [3 * C * (U + 2 * D) - V * X]\right] \end{array}\right\}} \tag{20}$$

$$\boxed{D_Mod_Ampl = A * K2 * C3} \tag{21}$$

The "erf" function is defined by: $\mathrm{erf}(x) = \frac{2}{\sqrt{\pi}} * \int_0^x e^{-t^2} dt$ (22)

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

COMMENTS

Global mathematical statement for the computation of the ocean echo model and partial derivatives

The mechanisms concerned by this definition are:

- GEN_MEC_MOD_01 - Ocean echo model and partial derivatives: Initialization (1)
- GEN_MEC_MOD_02 - Ocean echo model and partial derivatives: Initialization (2)
- GEN_MEC_MOD_03 - Ocean echo model and partial derivatives: Computation
- GEN_MEC_MOD_04 - Computation of "Gamma" and "Alpha" altimeter parameters

As described in AD4 (see "ALT_RET_OCE_01 - To perform the ocean-1 retracking" or "ALT_RET_OCE_02 - To perform the ocean-2 retracking"), the expression of the ocean echo model versus time is derived from Hayne's model (Hayne, 1980).

Assuming a gaussian Point Target Response, and accounting for the following variables:

- Time

t	"T"	in the specifications

- Instrumental features

θ_o : antenna beamwidth	"Ant_Beam"	in the specifications
σ_p : PTR width [(1)]	"PTR_Width"	in the specifications
h : mean satellite altitude	"Sat_Alt"	in the specifications

- Altimetric parameters

τ : epoch of the echo model	"Epoch"	in the specifications
σ_s : ocean surface roughness	"SigmaS"	in the specifications
σ_c : composite Sigma	"SigmaC"	in the specifications
P_u : amplitude	"Ampl"	in the specifications
P_n : thermal noise level	"TN"	in the specifications
ξ : mispointing (off-nadir angle)		
λ_s : skewness coefficient	"Skew"	in the specifications
$\lambda = \lambda_s \left(\frac{\sigma_s}{\sigma_c} \right)^3$	"L"	in the specifications
$\mu = \frac{\sigma_c^2 - \sigma_s^2}{\sigma_s^2 \, \sigma_c}$	"M"	in the specifications

(1) $\sigma_c^2 = \sigma_p^2 + \sigma_s^2$

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- Constants

c : velocity of light	"Light_Vel"	in the specifications
R_e : earth radius	"Earth_Rad"	in the specifications

- Instrumental-derived parameters

$\gamma = \frac{2}{\mathrm{Log}_e(2)}.\sin^2(\frac{\theta_0}{2})$	"Gamma"	in the specifications
$\alpha = \frac{4c}{\gamma h\left(1+\frac{h}{R_e}\right)}$	"Alpha"	in the specifications
$a_\xi = \exp\left(\frac{-4\sin^2\xi}{\gamma}\right)$	"A"	in the specifications
$b_\xi = \cos(2\xi) - \frac{\sin^2(2\xi)}{\gamma}$	"B"	in the specifications
$c_\xi = b_\xi \alpha$	"C"	in the specifications
$d_\xi = c_\xi \sigma_c$	"D"	in the specifications

- Time-dependent parameters

$u = \frac{t-\tau}{\sigma_c} - d_\xi$	"U"	in the specifications
$y = \frac{u}{\sqrt{2}}$	"Y"	in the specifications
$v = \frac{t-\tau}{\sigma_c^2} + c_\xi$	"V"	in the specifications
$x = 2u + 3d_\xi$	"X"	in the specifications

- Other parameters

$K_1 = a_\xi P_u$	"K1"	in the specifications
$K_2 = \exp\left[-d_\xi\left(u+\frac{d_\xi}{2}\right)\right]$	"K2"	in the specifications
$K = K_1 K_2$	"K"	in the specifications
$C_{11} = \frac{1}{2} * [1+\mathrm{erf}(y)]$	"C11"	in the specifications
$C_{12} = 1 + \frac{\lambda}{6} d_\xi^3$	"C12"	in the specifications
$C_1 = C_{11} C_{12}$	"C1"	in the specifications
$C_{21} = \frac{1}{\sqrt{2\pi}} \exp(-y^2)$	"C21"	in the specifications
$C_{22} = -\frac{\lambda}{6}\left[u^2 + 3d_\xi u + 3d_\xi^2 - 1\right]$	"C22"	in the specifications

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

$C_2 = C_{21}\, C_{22}$	"C2"	in the specifications
$C_3 = C_1 + C_2$	"C3"	in the specifications

The expression versus time of the echo model Vm(t) and of its partial derivatives with respect to the epoch (τ), composite Sigma (σ_c) and the amplitude (P_u) are given by:

$$Vm(t) = K\, C_3 + P_n$$

$$\frac{\partial Vm(t)}{\partial \tau} = \frac{K}{\sigma_c}\left\{ d_\xi C_3 + C_{21}\left[-C_{12} + \frac{\lambda}{6} x + C_{22} u \right] \right\}$$

$$\frac{\partial Vm(t)}{\partial \sigma_c} = K\left\{ c_\xi^2 \sigma_c C_3 + C_{11} \frac{\lambda d_\xi^2}{2} (\mu d_\xi + c_\xi) - C_{21}\left[C_{12} v - C_{22}(3\mu + uv) + \frac{\lambda}{6}\left[3c_\xi (u + 2d_\xi) - vx \right] \right] \right\}$$

$$\frac{\partial Vm(t)}{\partial P_u} = a_\xi\, K_2\, C_3$$

Implementation

- "Gamma" and "Alpha" parameters are computed in "GEN_MEC_MOD_04 - Computation of "Gamma" and "Alpha" altimeter parameters"
- In order to minimize the amount of operations within the ocean-1 and the ocean-2 retrackings and to optimize the data management within these algorithms, the echo model and its partial derivatives are computed from three algorithms:
  - "GEN_MEC_MOD_01 - Ocean echo model and partial derivatives: Initialization (1)", to compute parameters which do not depend on the echo samples (μ, λ, c_ξ, d_ξ, C_{12} and K_1)
  - "GEN_MEC_MOD_02 - Ocean echo model and partial derivatives: Initialization (2)", to compute time-dependent parameters for one sample (u, y and v)
  - "GEN_MEC_MOD_03 - Ocean echo model and partial derivatives: Computation" to compute the echo model and its partial derivatives for one sample

  and the computations of the echo model and partial derivatives are optimized for each part of the model (thermal noise plateau, leading edge, trailing edge).

REFERENCES

- Hayne G.S. 1980: "Radar Altimeter Mean Return Waveforms from Near-Normal-Incidence Ocean Surface Scattering". IEEE Trans. on antennas and propagation, Vol. AP-28, n°5, pp. 687-692

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_MOD_04 - Computation of "Gamma" and "Alpha" altimeter parameters

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J.P. DUMONT CLS	
Checked by:	For the JASON-1 SWT G. HAYNE For the JASON-1 Project P. VINCENT CNES S. DESAI JPL For the F-PAC / ENVISAT Project J. BENVENISTE ESA P. VINCENT CNES	
Approved by:	P. VINCENT CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To compute the "gamma" and "alpha" altimeter parameters from the antenna beamwidth and the satellite altitude.

ALGORITHM SPECIFICATION

Input data

- Antenna beamwidth : Ant_Beam (degree)
- Satellite altitude : Sat_Alt (m)
- Earth radius : Earth_Rad (m)
- Light velocity : Light_Vel (m/s)

Output data

- "Gamma" parameter : Gamma (/)
- "Alpha" parameter : Alpha (s^{-1})
- Execution status

Processing

$$\text{Gamma} = \frac{2}{\log_e(2)} * \sin^2\left(\frac{\text{Ant_Beam}}{2} * \frac{\pi}{180}\right) \quad (1)$$

$$\text{Alpha} = \frac{4 * \text{Light_Vel}}{\text{Gamma} * \text{Sat_Alt} * \left(1 + \frac{\text{Sat_Alt}}{\text{Earth_Rad}}\right)} \quad (2)$$

COMMENTS

A global mathematical for the computation of the ocean echo model and partial derivatives is given in section "Comments" of "GEN_MEC_MOD_03 - Ocean echo model and partial derivatives: Computation".

REFERENCES

None

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_MIS_01 - Ground distance between two points

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J. STUM	CLS	
Checked by:	N. PICOT	CNES	
Approved by:	P. VINCENT	CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To compute the ground distance between two points defined by their respective latitudes and longitudes.

ALGORITHM SPECIFICATION

Input data

- Latitude of the first point : Lat1 (degrees)
- Longitude of the first point : Lon1 (degrees)
- Latitude of the second point : Lat2 (degrees)
- Longitude of the second point : Lon2 (degrees)
- Characteristics of the reference ellipsoid:
  - Semi major axis : SM_Axis (m)
  - Flattening : Flattening (/)

Output data

- Distance between the two points : D (m)
- Execution status

Processing

- $\text{Sin_Lat1} = \sin\left(\text{Lat1} * \frac{\pi}{180}\right)$ (1)
- $\text{Sin_Lat2} = \sin\left(\text{Lat2} * \frac{\pi}{180}\right)$ (2)
- $\text{Cos_Lat1} = \cos\left(\text{Lat1} * \frac{\pi}{180}\right)$ (3)
- $\text{Cos_Lat2} = \cos\left(\text{Lat2} * \frac{\pi}{180}\right)$ (4)
- $\text{Cos_DifLon} = \cos\left[\left(\text{Lon1} - \text{Lon2}\right) * \frac{\pi}{180}\right]$ (5)
- $\text{CosD} = \text{Sin_Lat1} * \text{Sin_Lat2} + \text{Cos_Lat1} * \text{Cos_Lat2} * \text{Cos_DifLon}$ (6)
- $\text{D} = \text{Arc cos}\ (\text{CosD})$ (7)
- $\text{Ecc2} = \text{Flattening} * (2 - \text{Flattening})$ (8)

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- $\text{Earth_Radius} = \dfrac{\text{SM_Axis}}{\sqrt{1 - \text{Ecc2} * (\text{Sin_Lat1})^2}}$ (9)
- $D = D * \text{Earth_Radius}$ (10)

COMMENTS

None

REFERENCES

None

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_MIS_02 - Search of the closest corresponding altimeter point in a reference track

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J. STUM	CLS	
Checked by:	N. PICOT	CNES	
Approved by:	P. VINCENT	CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To find the closest corresponding altimeter point in a reference track

ALGORITHM SPECIFICATION

Input data

- Latitude of the altimeter point of the reference track : Lat_Ref (degrees)
- Longitude of the altimeter point of the reference track : Lon_Ref (degrees)
- Number of altimeter points of the given track : Nb_Giv (/)
- Index of the altimeter point of the given track used as a first guess for the search : I_Mes_First_Guess (/)
- Latitudes of the altimeter points of the given track : Lat_Alt[0:Nb_Giv-1] (degrees)
- Longitudes of the altimeter points of the given track : Lon_Alt[0:Nb_Giv-1] (degrees)
- Characteristics of the reference ellipsoid:
  - Semi major axis : SM_Axis (m)
  - Flattening : Flattening (/)

Output data

- Index of the real altimeter point of the given track,
  which is the closest from the altimeter point of the reference track : I_Mes_Closest (/)
- Distance Dmin between the altimeter point of the reference track
  and this real altimeter point : Dmin (m)
- Execution status

Processing

- I_Mes = I_Mes_First_Guess (1)
- Compute the distance Dmin between the altimeter point of the reference track and the altimeter point of the given track, using the mechanism “GEN_MEC_MIS_01 - Ground distance between two points”:

  The input parameters of which are:
  - Lat1 = Lat_Ref
  - Lat2 = Lat_Alt[I_Mes]
  - Lon1 = Lon_Ref
  - Lon2 = Lon_Alt[I_Mes]
  - SM_Axis
  - Flattening

  And the output parameter of which is:

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- Dmin

- Compute the distance D between the altimeter point of the reference track and the next altimeter point in the given track, using the mechanism "GEN_MEC_MIS_01 - Ground distance between two points":

  The input parameters of which are:

  - Lat1 = Lat_Ref
  - Lat2 = Lat_Alt[I_Mes+1]
  - Lon1 = Lon_Ref
  - Lon2 = Lon_Alt[I_Mes+1]
  - SM_Axis
  - Flattening

  And the output parameter of which is:

  - D

- If D ≥ Dmin, then :
  - I_Mes_Closest = I_Mes (2)

- Else :
  - While (D < Dmin), do the following processing:
    - DMin = D (3)
    - I_Mes = I_Mes + 1 (4)
    - Compute the distance D between the altimeter point of the reference track and the next altimeter point in the given track, using the mechanism "GEN_MEC_MIS_01 - Ground distance between two points":

      The input parameters of which are:

      - Lat1 = Lat_Ref
      - Lat2 = Lat_Alt[I_Mes+1]
      - Lon1 = Lon_Ref
      - Lon2 = Lon_Alt[I_Mes+1]
      - SM_Axis
      - Flattening

      And the output parameter of which is:

      - D
  - I_Mes_Closest = I_Mes – 1 (5)

COMMENTS

None

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

REFERENCES

None

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_MIS_03 - Location of missing measurements

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J. STUM	CLS	
Checked by:	N. PICOT	CNES	
Approved by:	P. VINCENT	CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

Altimeters are supposed to provide measurements approximately every 1 second. In some cases however, some expected measurements are missing, because of loss of tracking, or loss of telemetry transmission, or on board or on ground anomaly. This algorithm is aimed at determining the location of such missing measurements. A theoretical altimeter ground track pattern (with no gap) is used as a reference. The completeness of the actual (measured) altimeter ground track is checked relative to this reference pattern. The ouput of this analysis is the location of points in the reference pattern that are missing in the actual one, and the location of points in the reference pattern that exist in the actual one.

ALGORITHM SPECIFICATION

Input data

- Number of altimeter measurements : Nb_Mes (/)
- Latitudes of the altimeter measurements : Lat[0:Nb_Mes-1] (degrees)
- Longitudes of the altimeter measurements : Lon[0:Nb_Mes-1] (degrees)
- Ku-band ranges of the altimeter measurements : Range_Ku[0:Nb_Mes-1] (m)
- Number of theoretical altimeter measurements : Nb_Theo
- Index of the first theoretical altimeter measurement to consider : I_Theo_First
- Index of the last theoretical altimeter measurement to consider : I_Theo_Last
- Latitudes of the theoretical altimeter measurements : Lat_Theo[0:Nb_Theo-1] (degrees)
- Longitudes of the theoretical altimeter measurements : Lon_Theo[0:Nb_Theo-1] (degrees)
- Slope of the linear variation of the time interval between two consecutive altimeter measurements as function of altimeter range: : Dt_Dh (s/m)
- Conversion factor of time into distance units : Dt_DL (m/s)
- Mean value of altimeter range corresponding to the mean value of the time interval between two consecutive altimeter measurements: : Range_Mean (m)
- Characteristics of the reference ellipsoid:
  - Semi major axis : SM_Axis (m)
  - Flattening : Flattening (/)

Output data

- The missing theoretical measurements flags Flag_Miss[0:Nb_Theo-1] (/)
- The number of missing measurements Nb_Miss (/)
- The number of present measurements Nb_Pres (/)
- The execution status

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Processing

- Nb_Miss = 0 (1)
- Nb_Pres = 0 (2)
- I_Mes_First_Guess = 0 (3)
- For each theoretical altimeter point: (for I_Theo = I_Theo_First to I_Theo_Last):
  - Find the real altimeter point which is the closest from the theoretical point, using the mechanism "GEN_MEC_MIS_02 - Search of the closest corresponding altimeter point in a reference track",

    The input parameters of which are:

    * Lat_Ref = Lat_Theo[I_Theo]
    * Lon_Ref = Lon_Theo[I_Theo]
    * Nb_Giv = Nb_Mes
    * I_Mes_First_Guess
    * Lat_Alt[0:Nb_Giv-1] = Lat[0:Nb_Mes-1]
    * Lon_Alt[0:Nb_Giv-1] = Lon[0:Nb_Mes-1]
    * SM_Axis
    * Flattening

    And the output parameters of which are:

    * I_Mes_Closest : the index of the real altimeter point which is the closest from the theoretical point.
    * The distance Dmin between the altimeter point of the theoretical track and this real altimeter point: Dmin (m)
  - Set the index of the neighboring theoretical point to the following one, excepted if the actual theoretical point is the last one, in this last case, the neighboring point is the previous one:

    * If I_Theo = Nb_Theo-1, then

      I_Theo_N = I_Theo – 1 (4)

      Else

      I_Theo_N = I_Theo+1 (5)
  - Compute the distance, DTheo, between the two consecutive points on the theoretical track, using the mechanism "GEN_MEC_MIS_01 - Ground distance between two points" :

    The input parameters of which are:

    * Lat1 = Lat_Theo[I_Theo]
    * Lat2 = Lat_Theo[I_Theo_N]

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- Lon1 = Lon_Theo[I_Theo]
- Lon2 = Lon_Theo[I_Theo_N]
- SM_Axis
- Flattening

And the output parameter of which is:

- DTheo

– Compute the distance, D_N, between the closest real altimeter point and the neighboring point on the theoretical track, using the mechanism “GEN_MEC_MIS_01 - Ground distance between two points”:

The input parameters of which are:

- Lat1 = Lat_Theo[I_Theo_N]
- Lat2 = Lat[I_Mes_Closest]
- Lon1 = Lon_Theo[I_Theo_N]
- Lon2 = Lon[I_Mes_Closest]
- SM_Axis
- Flattening

And the output parameter of which is:

- D_N

– Compute the cosine of angle Alpha between the straight line defined by the two consecutive points on the theoretical track and the straight line defined by the first theoretical point and the input altimeter measurement:

$$\text{CosAlpha} = \frac{\text{Dmin}^2 + \text{DTheo}^2 - \text{D_N}^2}{2 * \text{Dmin} * \text{DTheo}} \quad (6)$$

– Add to the distance between the two consecutive points on the theoretical track a slight distance correction due to possible variation of the real distance with altimeter range (TOPEX heritage):

- Compute the variation of the time interval between two consecutive altimeter measurements, as a function of altimeter range:

$$\text{Delta_Time} = \text{Dt_Dh} * (\text{Range_Ku[I_Mes_Closest]} - \text{Range_Mean}) \quad (7)$$

- Translate this variation into distance units:

$$\text{Delta_D} = \text{Dt_DL} * \text{Delta_Time} \quad (8)$$

– If $\text{Dmin} * \text{ABS(CosAlpha)} > \frac{\text{DTheo} + \text{Delta_D}}{2}$, then (the altimeter point is missing in the theoretical track):

- Nb_Miss = Nb_Miss + 1 (9)
- Flag_Miss[I_Theo] = 1 (10)

Else (the altimeter point is present in the theoretical track):

- Nb_Pres = Nb_Pres + 1 (11)
- Flag_Miss[I_Theo] = 0 (12)

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

– I_Mes_First_Guess = I_Mes_Closest (13)

COMMENTS

None

REFERENCES

None

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_CRO_01 - Search of two consecutive measurements

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J. STUM	CLS	
Checked by:	N. PICOT	CNES	
Approved by:	P. VINCENT	CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To find two consecutive measurements (i.e., no time gap between these two measurements) in a series of measurements

ALGORITHM SPECIFICATION

Input data

- Number of input measurements : Nb_Mes (/)
- Time tags of the measurements : Time_Tag[0:Nb_Mes-1] (seconds)
- Index of measurement in the series from which search is beginning : I_Beg
- Index of measurement in the series at which search is ending : I_End
- Type of boundary searched in the data series [(1)] : Search_type (/)
- Time gap permitted between two consecutive measurements : Time_Gap (s) [(2)]

Output data

- Index of the first measurement in the series from which two consecutive measurements exist : I_Segm (/)
- Execution status

Processing

- I_Segm = I_Beg (1)
- If Search_Type ≤ 1, then:
  - Do while ((ABS(Time_Tag[I_Segm] – Time_Tag[I_Segm + 1]) > Time_Gap) AND (I_Segm < I_End)):
    I_Segm = I_Segm + 1 (2)

  Else:
  - Do while ((ABS(Time_Tag[I_Segm] – Time_Tag[I_Segm - 1]) > Time_Gap) AND (I_Segm > I_End)):
    I_Segm = I_Segm - 1 (3)

---

(1) 0 if the lower boundary is searched without knowing the upper boundary, 1 if the lower boundary is searched with the upper boundary being known, 2 if the upper boundary is searched

(2) 1.5 s if no gap permitted between two consecutive altimeter points

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

COMMENTS

None

REFERENCES

None

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_CRO_02 - Presence of a common latitude domain

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J. STUM	CLS	
Checked by:	N. PICOT	CNES	
Approved by:	P. VINCENT	CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To determine, from their latitude limits, if a latitude domain common to the two arcs exists or not.

ALGORITHM SPECIFICATION

Input data

- Latitude of the first measurement of arc 1 : Lat_First_Arc1 (degree)
- Latitude of the last measurement of arc 1 : Lat_Last_Arc1 (degree)
- Latitude of the first measurement of arc 2 : Lat_First_Arc2 (degree)
- Latitude of the last measurement of arc 2 : Lat_Last_Arc2 (degree)

Output data

- Flag of presence of a common latitude domain [1] : Flag_Lat (/)
- Execution status

Processing

- Latmin_Arc1 = MIN(Lat_First_Arc1, Lat_Last_Arc1) (1)
- Latmin_Arc2 = MIN(Lat_First_Arc2, Lat_Last_Arc2) (2)
- Latmax_Arc1 = MAX(Lat_First_Arc1, Lat_Last_Arc1) (3)
- Latmax_Arc2 = MAX(Lat_First_Arc2, Lat_Last_Arc2) (4)
- If ((Latmin_Arc2 > Latmax_Arc1) OR (Latmax_Arc2 < Latmin_Arc1)) then:

  Flag_Lat = 1 (5)

  Else:

  Flag_Lat = 0 (6)

COMMENTS

None

REFERENCES

None

---

[1] 0 if presence, 1 if absence

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_CRO_03 - Presence of a common longitude domain

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J. STUM	CLS	
Checked by:	N. PICOT	CNES	
Approved by:	P. VINCENT	CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To determine, from their longitude limits, if a longitude domain common to the two arcs exists or not.

ALGORITHM SPECIFICATION

Input data

- Longitude of the first measurement of arc 1 : Lon_First_Arc1 (degree)
- Longitude of the last measurement of arc 1 : Lon_Last_Arc1 (degree)
- Longitude of the first measurement of arc 2 : Lon_First_Arc2 (degree)
- Longitude of the last measurement of arc 2 : Lon_Last_Arc2 (degree)

Output data

- Flag of presence of a common longitude domain [1] : Flag_Lon (/)
- Greatest longitude among the arc 1 and arc2 first longitudes : Max_Lon_First_Arc1_Arc2 (degree)
- Smallest longitude among the arc 1 and arc2 last longitudes : Min_Lon_Last_Arc1_Arc2 (degree)
- Execution status

Processing

- Lonmin_Arc1 = MIN(Lon_First_Arc1, Lon_Last_Arc1) (1)
- Lonmin_Arc2 = MIN(Lon_First_Arc2, Lon_Last_Arc2) (2)
- Lonmax_Arc1 = MAX(Lon_First_Arc1, Lon_Last_Arc1) (3)
- Lonmax_Arc2 = MAX(Lon_First_Arc2, Lon_Last_Arc2) (4)
- Max_Lonmin_Arc1_Arc2 = MAX(Lonmin_Arc1, Lonmin_Arc2) (5)
- Min_Lonmax_Arc1_Arc2 = MIN(Lonmax_Arc1, Lonmax_Arc2) (6)
- If (Max_Lonmin_Arc1_Arc2 > Min_Lonmax_Arc1_Arc2) then:

  Flag_Lon = 1 (7)

  Else

  Flag_Lon = 0 (8)

[1] 0 if presence, 1 if absence

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

COMMENTS

None

REFERENCES

None

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_CRO_04 - Bisection

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J. STUM	CLS	
Checked by:	N. PICOT	CNES	
Approved by:	P. VINCENT	CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To determine, among a series of increasing longitudes, the closest smaller longitude from a given longitude, by bisection of the series.

ALGORITHM SPECIFICATION

Input data

- Number of points in the series : Nbmes
- Series of longitudes : Lon[0:Nbmes-1] (degrees)
- The given longitude : Lon_Giv (degrees)
- First index of the series from which bisection is performed : I_Min (/)
- Last index of the series from which bisection is performed : I_Max (/)

Output data

- Index corresponding to the closest smaller longitude of the series : I_Clos
- Execution status

Processing

- I_Beg = I_Min (1)
- I_End = I_Max (2)
- For I = 0 to 100:
  - I_Clos = $\frac{I_Beg + I_End}{2}$ (3)
  - If ((I_End – I_Beg) ≤ 1) then:

    exit
  - If (Lon[I_Clos] > Lon_Giv) then:

    I_End = I_Clos (4)

    Else

    I_Beg = I_Clos (5)
- If I > 100, then:

  Set the execution status to “no convergence in GEN_MEC_CRO_04”

COMMENTS

None

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

REFERENCES

None

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_CRO_05 - Change of the sign of the latitude difference

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J. STUM	CLS	
Checked by:	N. PICOT	CNES	
Approved by:	P. VINCENT	CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To determine if, within the common longitude domain, the difference of the beginning latitudes of the two arcs and the difference of the end latitudes of the two arcs have an opposite sign or not. If they have an opposite sign, then the crossing of the two arcs is possible within this longitude domain.

ALGORITHM SPECIFICATION

Input data

- Latitude of the first measurement of arc 1 : Lat_First_Arc1 (degree)
- Latitude of the last measurement of arc 1 : Lat_Last_Arc1 (degree)
- Latitude of the first measurement of arc 2 : Lat_First_Arc2 (degree)
- Latitude of the last measurement of arc 2 : Lat_Last_Arc2 (degree)

Output data

- Flag of possibility of arcs crossing [1] : Flag_Lat (/)
- Execution status

Processing

- If (((Lat_First_Arc2 > Lat_First_Arc1) AND (Lat_Last_Arc2 > Lat_Last_Arc1))

  OR ((Lat_First_Arc1 > Lat_First_Arc2) AND (Lat_Last_Arc1 > Lat_Last_Arc2))) then:

  – Flag_Lat = 1 (1)

  Else:

  – Flag_Lat = 0 (2)

COMMENTS

None

REFERENCES

None

[1] 0 if crossing is possible, 1 if no crossing is possible

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_CRO_06 - Search of one boundary of the common longitude domain

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J. STUM	CLS	
Checked by:	N. PICOT	CNES	
Approved by:	P. VINCENT	CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To determine one of the boundaries of the longitude domain which is common to the two arcs. This boundary is given by the two arc segments of arc 1 and arc 2 (an arc segment is made of two consecutive altimeter measurements) for which the two longitudes of one segment bracket the beginning longitude of the other segment.

ALGORITHM SPECIFICATION

Input data

- Number of altimeter measurements in arc 1 : Nb_Mes1 (/)
- Altimeter time tags of arc 1 : Time1[0:Nb_Mes1-1] (1)
- Altimeter latitudes of arc 1 : Lat1[0:Nb_Mes1-1] (degree)
- Altimeter longitudes of arc 1 : Lon1[0:Nb_Mes1-1] (degree)
- First boundary in latitude of arc 1 : Lat_First1 (degree)
- Second boundary in latitude of arc 1 : Lat_Last1 (degree)
- First boundary in longitude of arc 1 : Lon_First1 (degree)
- Second boundary in longitude of arc 1 : Lon_Last1 (degree)
- First index to consider in arc 1 for the search : I_First1
- Last index to consider in arc 1 for the search : I_Last1
- Number of altimeter measurements in arc 2 : Nb_Mes2 (/)
- Altimeter time tags of arc 2 : Time2[0:Nb_Mes2-1] (1)
- Altimeter latitudes of arc 2 : Lat2[0:Nb_Mes2-1] (degree)
- Altimeter longitudes of arc 2 : Lon2[0:Nb_Mes2-1] (degree)
- First boundary in latitude of arc 2 : Lat_First2 (degree)
- Second boundary in latitude of arc 2 : Lat_Last2 (degree)
- First boundary in longitude of arc 2 : Lon_First2 (degree)
- Second boundary in longitude of arc 2 : Lon_Last2 (degree)
- First index to consider in arc 2 for the search : I_First2 (/)
- Last index to consider in arc 2 for the search : I_Last2 (/)
- The type of boundary searched in the data series (2) : Search_type

(1) Seconds elapsed since 01/01/1950 0 h.

(2) 0 if the lower boundary is searched without knowing the upper boundary, 1 if the lower boundary is searched with the upper boundary being known, 2 if the upper boundary is searched

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- Time gap permitted between two consecutive measurements on arc 1 : Time_Gap1 [3] (s)
- Time gap permitted between two consecutive measurements on arc 2 : Time_Gap2 [3](s)

Output data

- The common boundary in longitude : Lon_Bound (degree)
- The latitude of arc 1 corresponding to the common boundary in longitude : Lat_Bound1 (degree)
- The latitude of arc 2 corresponding to the common boundary in longitude : Lat_Bound2 (degree)
- The index of the measurement of a segment of arc 1 which has a common longitude domain with a segment of arc 2 : I_Closest_Segm1 (/)
- The index of the measurement of a segment of arc 2 which has a common longitude domain with a segment of arc 1 : I_Closest_Segm2 (/)
- The index of the measurement of arc 2 which longitude is the closest and smaller than the longitude of arc 1 : I_Closest_Meas2 (/)
- The flag for absence of common latitude domain for the two arcs : Flag_Lat [4]
- The flag for absence of common longitude domain for the two arcs : Flag_Lon [4]
- The flag for absence of arc segments on arc 1 : Flag_Segm1 [4]
- The flag for absence of arc segments on arc 2 : Flag_Segm2 [4]
- The flag for absence of a time gap on arc 1 after the closest longitude if found on arc 1 [5] : Flag_Gap1
- The flag for absence of a time gap on arc 2 after the closest longitude if found on arc 2 [5] : Flag_Gap2
- Execution status

Processing

- If Search_Type ≤ 1, then:

  I_Beg1 = I_First1 (1)

  I_End1 = I_Last1 (2)

---

[3] 1.5 s if no gap permitted between two consecutive altimeter points

[4] 1 = absence

[5] 0 = absence, 1 = presence

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

I_Beg2 = I_First2 (3)

I_End2 = I_Last2 (4)

Lat_End1 = Lat_Last1 (5)

Lat_Beg2 = Lat_First2 (6)

Lat_End2 = Lat_Last2 (7)

Lon_End1 = Lon_Last1 (8)

Lon_Beg2 = Lon_First2 (9)

Lon_End2 = Lon_Last2 (10)

- Else:

  I_Beg1 = I_Last1 (11)

  I_End1 = I_First1 (12)

  I_Beg2 = I_Last2 (13)

  I_End2 = I_First2 (14)

  Lat_End1 = Lat_First1 (15)

  Lat_Beg2 = Lat_Last2 (16)

  Lat_End2 = Lat_First2 (17)

  Lon_End1 = Lon_First1 (18)

  Lon_Beg2 = Lon_Last2 (19)

  Lon_End2 = Lon_First2 (20)

- Flag_Gap1 = 1 (21)
- Flag_Gap2 = 1 (22)
- Flag_Segm1 = 0 (23)
- Flag_Segm2 = 0 (24)
- I_Closest_Meas2 = I_Last2 (25)
- The presence of a latitude domain common to the two arcs is determined, using the mechanism “GEN_MEC_CRO_02 - Presence of a common latitude domain”,

  the input parameters of which are:

  – Lat_First_Arc1 = Lat1[I_Beg1]
  – Lat_Last_Arc1 = Lat_End1
  – Lat_First_Arc2 = Lat_Beg2
  – Lat_Last_Arc2 = Lat_End2

  and the output parameters of which are:

  – Flag_Lat (= 0 if a common latitude domain exists, = 1 if no common latitude domain exists)

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- The execution status

- If Flag_Lat = 1, then:
  - Exit
- The presence of a longitude domain common to the two arcs is determined, using the mechanism "GEN_MEC_CRO_03 - Presence of a common longitude domain ",

  the input parameters of which are:
  - Lon_First_Arc1 = Lon1[I_Beg1]
  - Lon_Last_Arc1 = Lon_End1
  - Lon_First_Arc2 = Lon_Beg2
  - Lon_Last_Arc2 = Lon_End2

  and the output parameters of which are:
  - Max_Lonmin_Arc1_Arc2
  - Min_Lonmax_Arc1_Arc2
  - Flag_Lon (= 0 if a common longitude domain exists, = 1 if no common longitude domain exists)
  - The execution status
- If Flag_Lon = 1, then:
  - Exit
- The first existing arc segment after I_Beg1 is searched on arc 1, using the mechanism "GEN_MEC_CRO_01 - Search of two consecutive measurements",

  the input parameters of which are:
  - Nb_Mes = Nb_Mes1
  - Time_Tag[0:Nb_Mes-1] = Time1[0:Nb_Mes1-1]
  - I_Beg = I_Beg1
  - I_End = I_End1
  - Search_Type
  - Time_Gap = Time_Gap1

  and the output parameters of which are:
  - I_Segm = I_Closest_Segm1
  - The execution status
- If I_Closest_Segm1 = I_End1, then:
  - Flag_Segm1 = 1 (no arc segment found on arc 1 after I_Beg1) (26)
  - Exit

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- If I_Closest_Segm1 ≠ I_Beg1, then:
  - The presence of a latitude domain common to the two arcs is determined, using the mechanism "GEN_MEC_CRO_02 - Presence of a common latitude domain",

    the input parameters of which are:
    - Lat_First_Arc1 = Lat1[I_Closest_Segm1]
    - Lat_Last_Arc1 = Lat_End1
    - Lat_First_Arc2 = Lat_Beg2
    - Lat_Last_Arc2 = Lat_End2

    and the output parameters of which are:
    - Flag_Lat (= 0 if a common latitude domain exists, = 1 if no common latitude domain exists)
    - The execution status
  - If Flag_Lat = 1, then:
    - Exit
  - The presence of a longitude domain common to the two arcs is determined, using the mechanism "GEN_MEC_CRO_03 - Presence of a common longitude domain ",

    the input parameters of which are:
    - Lon_First_Arc1 = Lon1[I_Closest_Segm1]
    - Lon_Last_Arc1 = Lon_End1
    - Lon_First_Arc2 = Lon_Beg2
    - Lon_Last_Arc2 = Lon_End2

    and the output parameters of which are:
    - Max_Lonmin_Arc1_Arc2
    - Min_Lonmax_Arc1_Arc2
    - Flag_Lon (= 0 if a common longitude domain exists, = 1 if no common longitude domain exists)
    - The execution status
  - If Flag_Lon = 1, then:
    - Exit
- If Search_Type ≠ 1, then:
  - The first existing arc segment after I_Beg2 is searched on arc 2, using the mechanism "GEN_MEC_CRO_01 - Search of two consecutive measurements",

    the input parameters of which are:
    - Nb_Mes = Nb_Mes2
    - Time_Tag[0:Nb_Mes-1] = Time2[0:Nb_Mes2-1]

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- I_Beg = I_Beg2
- I_End = I_End2
- Search_Type
- Time_Gap = Time_Gap2

and the output parameters of which are:

- I_Segm = I_Closest_Segm2
- The execution status

– If I_Closest_Segm2 = I_End2, then:

- Flag_Segm2 = 1 (no arc segment found on arc 2 after I_Beg2) (27)
- Exit

– If I_Closest_Segm2 ≠ I_Beg2, then:

- The presence of a latitude domain common to the two arcs is determined, using the mechanism "GEN_MEC_CRO_02 - Presence of a common latitude domain",

  the input parameters of which are:

  - Lat_First_Arc1 = Lat1[I_Closest_Segm1]
  - Lat_Last_Arc1 = Lat_End1
  - Lat_First_Arc2 = Lat2[I_Closest_Segm2]
  - Lat_Last_Arc2 = Lat_End2

  and the output parameters of which are:

  - Flag_Lat (= 0 if a common latitude domain exists, = 1 if no common latitude domain exists)
  - The execution status

- If Flag_Lat = 1, then:

  - Exit

- The presence of a longitude domain common to the two arcs is determined, using the mechanism "GEN_MEC_CRO_03 - Presence of a common longitude domain ",

  the input parameters of which are:

  - Lon_First_Arc1 = Lon1[I_Closest_Segm1]
  - Lon_Last_Arc1 = Lon_End1
  - Lon_First_Arc2 = Lon2[I_Closest_Segm2]
  - Lon_Last_Arc2 = Lon_End2

  and the output parameters of which are:

  - Max_Lonmin_Arc1_Arc2
  - Min_Lonmax_Arc1_Arc2

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- ◊ Flag_Lon (= 0 if a common longitude domain exists, = 1 if no common longitude domain exists)
- ◊ The execution status

∗ If Flag_Lon = 1, then:

- ◊ Exit

- Else (Search_Type = 1)

  I_Closest_Segm2 = I_First2 (28)

- If Search_Type ≤ 1, then:

  Lon_Bound = Max_Lonmin_Arc1_Arc2 (29)

- Else (Search_Type = 2):

  Lon_Bound = Min_Lonmax_Arc1_Arc2 (30)

- Do while ((Flag_Gap1 = 1) AND (Flag_Gap2 = 1))

  – If Lon_Bound = Lon1[I_Closest_Segm1], then:

  ∗ Search on arc 2 of the closest longitude smaller than Lon1[I_Closest_Segm1], using the mechanism "GEN_MEC_CRO_07 - Search on one arc of the closest longitude smaller than a given longitude of the other arc",

  the input parameters of which are:

  - ◊ Nb_Mes_Arc = Nb_Mes2
  - ◊ Time_Arc[0:Nb_Mes_Arc-1] = Time2[0:Nb_Mes2-1]
  - ◊ Lat_Arc[0:Nb_Mes_Arc-1] = Lat2[0:Nb_Mes2-1]
  - ◊ Lon_Arc[0:Nb_Mes_Arc-1] = Lon2[0:Nb_Mes2-1]
  - ◊ I_Segm_Arc = I_Closest_Segm2
  - ◊ I_End_Arc = I_End2
  - ◊ Lat_End_Arc = Lat_End2
  - ◊ Lon_End_Arc = Lon_End2
  - ◊ Lat_Beg_Other_Arc = Lat1[I_Closest_Segm1]
  - ◊ Lat_End_Other_Arc = Lat_End1
  - ◊ Lon_Beg_Other_Arc = Lon1[I_Closest_Segm1]
  - ◊ Lon_End_Other_Arc = Lon_End1
  - ◊ Search_Type
  - ◊ Time_Gap_Arc = Time_Gap2

  and the output parameters of which are:

  - ◊ I_Closest_Segm_Arc = I_Closest_Segm2
  - ◊ I_Closest_Meas_Arc = I_Closest_Meas2

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- ◊ Lon_Bound_New = Lon_Bound
- ◊ Lat_Bound_New = Lat_Bound2
- ◊ Flag_Gap_Arc = Flag_Gap2
- ◊ Flag_Lat
- ◊ Flag_Lon
- ◊ Flag_Segm = Flag_Segm2

∗ Lat_Bound1 = Lat1(I_Closest_Segm1) (31)

∗ If Flag_Lat = 1, OR Flag_Lon = 1, OR Flag_Segm2 = 1, then:

- ◊ exit

– Else (Lon_Bound = Lon2[I_Closest_Segm2]):

∗ Search on arc 1 of the closest longitude smaller than Lon2[I_Closest_Segm2], using the mechanism "GEN_MEC_CRO_07 - Search on one arc of the closest longitude smaller than a given longitude of the other arc",

the input parameters of which are:

- ◊ Nb_Mes_Arc = Nb_Mes1
- ◊ Time_Arc[0:Nb_Mes_Arc-1] = Time1[0:Nb_Mes1-1]
- ◊ Lat_Arc[0:Nb_Mes_Arc-1] = Lat1[0:Nb_Mes1-1]
- ◊ Lon_Arc[0:Nb_Mes_Arc-1] = Lon1[0:Nb_Mes1-1]
- ◊ I_Segm_Arc = I_Closest_Segm1
- ◊ I_End_Arc = I_End1
- ◊ Lat_End_Arc = Lat_End1
- ◊ Lon_End_Arc = Lon_End1
- ◊ Lat_Beg_Other_Arc = Lat2[I_Closest_Segm2]
- ◊ Lat_End_Other_Arc = Lat2[I_End2]
- ◊ Lon_Beg_Other_Arc = Lon2[I_Closest_Segm2]
- ◊ Lon_End_Other_Arc = Lon2[I_End2]
- ◊ Search_Type
- ◊ Time_Gap_Arc = Time_Gap1

and the output parameters of which are:

- ◊ I_Closest_Segm_Arc = I_Closest_Segm1
- ◊ I_Closest_Meas_Arc = I_Closest_Meas1
- ◊ Lon_Bound_New = Lon_Bound
- ◊ Lat_Bound_New = Lat_Bound1

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- ◊ Flag_Gap_Arc = Flag_Gap1
- ◊ Flag_Lat
- ◊ Flag_Lon
- ◊ Flag_Segm = Flag_Segm1

∗ Lat_Bound2 = Lat2(I_Closest_Segm2) (32)

∗ If Flag_Lat = 1, OR Flag_Lon = 1, OR Flag_Segm1 = 1, then:

- ◊ exit

COMMENTS

None

REFERENCES

None

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_CRO_07 - Search on one arc of the closest longitude smaller than a given longitude of the other arc

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J. STUM	CLS	
Checked by:	N. PICOT	CNES	
Approved by:	P. VINCENT	CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To search on one arc the closest longitude smaller than a given longitude which belongs to the other arc.

ALGORITHM SPECIFICATION

Input data

- Number of altimeter measurements in the input arc : Nb_Mes_Arc (/)
- Altimeter time tags of the input arc (seconds elapsed since 01/01/1950 0h.) : Time_Arc[0:Nb_Mes_Arc-1]
- Altimeter latitudes of the input arc (degree) : Lat_Arc[0:Nb_Mes_Arc-1]
- Altimeter longitudes of the input arc (degree) : Lon_Arc[0:Nb_Mes_Arc-1]
- Index of the first segment to consider in the input arc for the search : I_Segm_Arc (/)
- Last index to consider in the arc for the search : I_End_Arc (/)
- End boundary in latitude for the arc (degree) : Lat_End_Arc
- End boundary in longitude for the arc (degree) : Lon_End_Arc
- Beginning latitude of the other arc (degree) : Lat_Beg_Other_Arc
- End latitude of the other arc (degree) : Lat_End_Other_Arc
- Beginning longitude of the other arc (degree) : Lon_Beg_Other_Arc
- End longitude of the other arc (degree) : Lon_End_Other_Arc
- The type of arc boundary searched in the data series [1] : Search_type (/)
- Time gap permitted between two consecutive measurements on the input arc: Time_Gap_Arc [2] (s)

Output data

- Index of the measurement of the segment of the input arc which brackets the given longitude of the other arc, if any, else of the measurement of the first segment of the input arc which is found after the given longitude of the other arc : I_Closest_Segm_Arc (/)

---

[1] 0 if the lower boundary is searched without knowing the upper boundary, 1 if the lower boundary is searched with the upper boundary being known, 2 if the upper boundary is searched

[2] 1.5 s if no gap permitted between two consecutive altimeter points

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- Index of the measurement of the input arc which longitude is the closest than the longitude of the other arc. This longitude is smaller than the longitude of the other arc if Search_Type ≤ 1, and greater than the longitude of the other arc if Search_Type = 2 : I_Closest_Meas_Arc (/)
- New longitude boundary of the two arcs: : Lon_Bound_New (degree)
- New latitude boundary of the input arc: : Lat_Bound_New (degree)
- Flag for absence of a time gap after the closest longitude found on the input arc (= 0 : absence, = 1 : presence) : Flag_Gap_Arc
- Flag for absence of common latitude domain for the two arcs (= 0 : presence, = 1 : absence) : Flag_Lat
- Flag for absence of common longitude domain for the two arcs (= 0 : presence, = 1 : absence) : Flag_Lon
- Flag for absence of arc segment after I_Closest_Arc (= 0 : presence, = 1 : absence) : Flag_Segm
- Execution status

Processing

- If Search_Type ≤ 1, then:

  I1 = I_Segm_Arc (1)

  I2 = I_End_Arc (2)
- Else (Search_Type = 2)

  I1 = I_End_Arc (3)

  I2 = I_Segm_Arc (4)
- The closest longitude smaller than the beginning longitude of the other arc is searched on the input arc, using the mechanism “GEN_MEC_CRO_04 - Bisection”,

  the input parameters of which are:

  – Nbmes = Nb_Mes_Arc
  – Lon[0:Nbmes-1] = Lon_Arc[0:Nb_Mes_Arc-1]
  – Lon_Giv = Lon_Beg_Other_Arc
  – I_Min = I1
  – I_Max = I2

  and the output parameters of which are:

  – I_Clos = I_Closest_Meas_Arc
  – The execution status

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- If Search_Type = 2, then:

  I_Closest_Meas_Arc = I_Closest_Meas_Arc + 1 (5)

- The first existing arc segment after I_Closest_Meas_Arc is searched on the input arc, using the mechanism "GEN_MEC_CRO_01 - Search of two consecutive measurements",

  the input parameters of which are:

  – Nb_Mes = Nb_Mes_Arc
  – Time_Tag[0:Nb_Mes-1] = Time_Arc[0:Nb_Mes_Arc-1]
  – I_Beg = I_Closest_Meas_Arc
  – I_End = I_End_Arc
  – Search_Type
  – Time_Gap = Time_Gap_Arc

  and the output parameters of which are:

  – I_Segm = I_Closest_Segm_Arc
  – The execution status

- If I_Closest_Segm_Arc = I_End_Arc, then:
  – Flag_Segm = 1 (no arc segment found after I_Closest_Meas_Arc) (6)
  – Exit

- If I_Closest_Segm_Arc = I_Closest_Meas_Arc, then:
  – Flag_Gap_Arc = 0 (no gap has been found, i.e., the closest longitude found by bisection is the beginning of an arc segment) (7)
  – Lon_Bound_New = Lon_Beg_Other_Arc (8)
  – If Search_Type ≤ 1, then:
    ∗ I_Segm_Beg = I_Closest_Segm_Arc (9)
    ∗ I_Segm_End = I_Closest_Segm_Arc + 1 (10)
  – Else (Search_type = 2):
    ∗ I_Segm_Beg = I_Closest_Segm_Arc – 1 (11)
    ∗ I_Segm_End = I_Closest_Segm_Arc (12)
  – Computing on the input arc the exact latitude Lat_Bound_New, corresponding to Lon_Beg_Other_Arc, by linear interpolation in latitude:
    ∗ First, computing the weights of linear interpolation, using mechanism "GEN_MEC_INT_01 - Linear weighting",

      the input parameters of which are:

      ◊ Xmin = Lon_Arc[I_Segm_Beg]

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

◊ Xmax = Lon_Arc[I_Segm_End]

◊ Xstep = Lon_Arc[I_Segm_End] – Lon_Arc[I_Segm_Beg]

◊ X = Lon_Beg_Other_Arc

and the output parameters of which are:

◊ Index_1 (unused)

◊ Index_2 (unused)

◊ W1 = w1

◊ W2 = w2

◊ The execution status

∗ Next, computing by linear interpolation in latitude the exact latitude corresponding to Lon_Beg_Other_Arc, using mechanism "GEN_MEC_INT_02 - Linear interpolation",

the input parameters of which are:

◊ X1 = Lat_Arc[I_Segm_Beg]

◊ X2 = Lat_Arc[I_Segm_End]

◊ W1 = w1

◊ W2 = w2

and the output parameters of which are:

◊ Lat_Bound_New

◊ The execution status

– If Search_Type ≥ 1, then:

∗ Looking at the possibility of arcs crossing within the common longitude domain, using the mechanism “GEN_MEC_CRO_05 - Change of the sign of the latitude difference”,

the input parameters of which are:

◊ Lat_First_Arc1 = Lat_Beg_Other_Arc

◊ Lat_Last_Arc1 = Lat_End_Other_Arc

◊ Lat_First_Arc2 = Lat_Bound_New

◊ Lat_Last_Arc2 = Lat_End_Arc

and the output parameters of which are:

◊ Flag_Lat

◊ The execution status

∗ If Flag_Lat = 1, then

◊ exit

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- Else (Search_Type = 0)
  - The presence of a latitude domain common to the two arcs is determined, using the mechanism "GEN_MEC_CRO_02 - Presence of a common latitude domain",

    the input parameters of which are:
    - Lat_First_Arc1 = Lat_Beg_Other_Arc
    - Lat_Last_Arc1 = Lat_End_Other_Arc
    - Lat_First_Arc2 = Lat_Bound_New
    - Lat_Last_Arc2 = Lat_End_Arc

    and the output parameters of which are:
    - Flag_Lat (= 0 if a common latitude domain exists, = 1 if no common latitude domain exists)
    - The execution status
  - If Flag_Lat = 1, then:
    - Exit
  - The presence of a longitude domain common to the two arcs is determined, using the mechanism "GEN_MEC_CRO_03 - Presence of a common longitude domain ",

    the input parameters of which are:
    - Lon_First_Arc1 = Lon_Beg_Other_Arc
    - Lon_Last_Arc1 = Lon_End_Other_Arc
    - Lon_First_Arc2 = Lon_Beg_Other_Arc
    - Lon_Last_Arc2 = Lon_End_Arc

    and the output parameters of which are:
    - Max_Lonmin_Arc1_Arc2
    - Min_Lonmax_Arc1_Arc2
    - Flag_Lon (= 0 if a common longitude domain exists, = 1 if no common longitude domain exists)
    - The execution status

- Else (I_Closest_Segm_Arc ≠ I_Closest_Meas_Arc):
  - Flag_Gap_Arc = 1 (a gap has been found, i.e., the closest longitude found by bisection is not the beginning of an arc segment) (13)
  - Lon_Bound_New = Lon_Arc[I_Closest_Segm_Arc] (14)
  - Lat_Bound_New = Lat_Arc[I_Closest_Segm_Arc] (15)

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- The presence of a latitude domain common to the two arcs is determined, using the mechanism "GEN_MEC_CRO_02 - Presence of a common latitude domain",

  the input parameters of which are:

  * Lat_First_Arc1 = Lat_Beg_Other_Arc
  * Lat_Last_Arc1 = Lat_End_Other_Arc
  * Lat_First_Arc2 = Lat_Arc[I_Closest_Segm_Arc]
  * Lat_Last_Arc2 = Lat_End_Arc

  and the output parameters of which are:

  * Flag_Lat (= 0 if a common latitude domain exists, = 1 if no common latitude domain exists)
  * The execution status

- If Flag_Lat > 0, then:

  * Exit

- The presence of a longitude domain common to the two arcs is determined, using the mechanism "GEN_MEC_CRO_03 - Presence of a common longitude domain ",

  the input parameters of which are:

  * Lon_First_Arc1 = Lon_Beg_Other_Arc
  * Lon_Last_Arc1 = Lon_End_Other_Arc
  * Lon_First_Arc2 = Lon_Arc[I_Closest_Segm_Arc]
  * Lon_Last_Arc2 = Lon_End_Arc

  and the output parameters of which are:

  * Max_Lonmin_Arc1_Arc2
  * Min_Lonmax_Arc1_Arc2
  * Flag_Lon (= 0 if a common longitude domain exists, = 1 if no common longitude domain exists)
  * The execution status

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_CRO_08 - Search of the intersecting segments

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J. STUM	CLS	
Checked by:	N. PICOT	CNES	
Approved by:	P. VINCENT	CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To determine the segment of arc 1 and the segment of arc 2 which do intersect, among a series of (a few, typically three) segments of each arc. The geometry is shown on **Figure 1**, where [A1,B1] is a segment of arc 1, and [A2,B2] is a segment of arc 2. Point M is the intersection of the two segments, which coordinates (latitude and longitude) are to be solved. The following vectorial equations can be written :

- $\overrightarrow{A_1M} = L1 * \overrightarrow{A_1B_1}$ (1)
- $\overrightarrow{A_2M} = L2 * \overrightarrow{A_2B_2}$ (2)

where L1 and L2 are the unknowns.

- $\overrightarrow{A_1A_2} = \overrightarrow{A_1M} - \overrightarrow{A_2M}$ (3)
- $L1 * \overrightarrow{A_1B_1} - L2 * \overrightarrow{A_2B_2} = \overrightarrow{A1A2}$ (4)

This last vectorial equation leads to the following linear system of two equations with the two unknowns L1 and L2 :

- L1 x [Lon1(B1) – Lon1(A1)] – L2 x [Lon2(B2) – Lon2(A2)] = Lon2(A2) – Lon1(A1) (5)
- L1 x [Lat1(B1) – Lat1(A1)] – L2 x [Lat2(B2) – Lat2(A2)] = Lat2(A2) – Lat1(A1) (6)

The crossover is located on the segment of arc 2 for which the values of L1 and L2 are both between 0 and 1. If the segments do not correspond to consecutive measurements, the crossover point is rejected.

The crossover longitude and latitude are then given by :

- Lon_Cros = Lon1(A1) + L1 x [Lon1(B1) – Lon1(A1)] (7)
- Lat_Cros = Lat1(A1) + L1 x [Lat1(B1) – Lat1(A1)] (8)

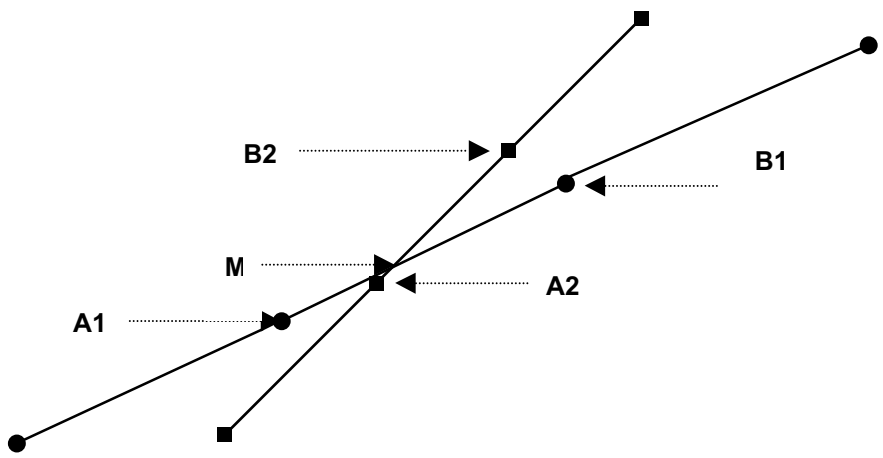


Figure 1

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

ALGORITHM SPECIFICATION

Input data

- Number of altimeter measurements in arc 1 : Nb_Mes1 (/)
- Altimeter time tags of arc 1 : Time1[0:Nb_Mes1-1] (1)
- Altimeter latitudes of arc 1 : Lat1[0:Nb_Mes1-1] (degree)
- Altimeter longitudes of arc 1 : Lon1[0:Nb_Mes1-1] (degree)
- First arc segment to consider in arc 1 for the search : I_Segm1_First
- Last arc segment to consider in arc 1 for the search : I_Segm1_Last
- Number of altimeter measurements in arc 2 : Nb_Mes2 (/)
- Altimeter time tags of arc 2 : Time2[0:Nb_Mes2-1] (1)
- Altimeter latitudes of arc 2 : Lat2[0:Nb_Mes2-1] (degree)
- Altimeter longitudes of arc 2 : Lon2[0:Nb_Mes2-1] (degree)
- First arc segment to consider in arc 2 for the search : I_Segm2_First
- Last arc segment to consider in arc 2 for the search : I_Segm2_Last
- Time gap permitted between two consecutive measurements on arc 1 : Time_Gap1 (2) (s)
- Time gap permitted between two consecutive measurements on arc 2 : Time_Gap2 (2)(s)

Output data

- Latitude of the crossover point : Lat_Cros
- Longitude of the crossover point : Lon_Cros
- Time tag of the crossover point on arc 1 : Time_Cros1
- Time-tag of the crossover point on arc 2 : Time_Cros2
- Index of the first measurement of the arc segment of arc 1 which crosses over the segment of arc 2 : I_Cros1 (/)
- Index of the first measurement of the arc segment of arc 2 which crosses over the segment of arc 1 : I_Cros2 (/)
- Weight for linear interpolation of the arc 1 parameters at crossover point : L1
- Weight for linear interpolation of the arc 2 parameters at crossover point : L2
- Flag for absence of a crossover : Flag_Cros (3)

---

(1) Seconds elapsed since 01/01/1950 0 h.

(2) = 1.5 s if no gap permitted between consecutive altimeter measurements

(3) = 1 for absence, = 0 for presence

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- Execution status

Processing

- For I = I_Segm1_First to I_Segm1_Last
  - If I ≥ 0 and I < Nb_Mes1-1, then :
    - ∗ Lon_A1B1 = Lon1[I+1] – Lon1[I] (1)
    - ∗ Lat_A1B1 = Lat1[I+1] – Lat1[I] (2)
    - ∗ For J = I_Segm2_First to I_Segm2_Last
      - ◊ If J ≥ 0 and J < Nb_Mes2-1, then :
        - ⇒ Lon_A2B2 = Lon2[J+1] – Lon2[J] (3)
        - ⇒ Lon_A1A2 = Lon2[J] – Lon1[I] (4)
        - ⇒ Lat_A2B2 = Lat2[J+1] – Lat2[J] (5)
        - ⇒ Lat_A1A2 = Lat2[J] – Lat1[I] (6)
        - ⇒ Determinant = Lat_A1B1 x Lon_A2B2 – Lon_A1B1 x Lat_A2B2 (7)
        - ⇒ Determ_L1 = Lat_A1A2 x Lon_A2B2 – Lon_A1A2 x Lat_A2B2 (8)
        - ⇒ Determ_L2 = Lon_A1B1 x Lat_A1A2 – Lat_A1B1 x Lon_A1A2 (9)
        - ⇒ L1 = Determ_L1/Determinant (10)
        - ⇒ L2 = Determ_L2/Determinant (11)
        - ⇒ If ((0 ≤ L1 ≤ 1) AND (0 ≤ L2 ≤ 1)) then :
          - → I_Cros1 = I (12)
          - → I_Cros2 = J (13)
          - → If ABS(Time1[I_Cros1+1] – Time1[I_Cros1]) > Time_Gap1, then :
            Flag_Cros = 1 (14)
            exit
          - → If ABS(Time2[I_Cros2+1] – Time2[I_Cros2]) > Time_Gap2, then :
            Flag_Cros = 1 (15)
            exit
          - → Lon_Cros = Lon1[I_Cros1] + L1 x Lon_A1B1 (16)
          - → Lat_Cros = Lat1[I_Cros1] + L1 x Lat_A1B1 (17)
          - → Time_Cros1 = Time1[I_Cros1] + L1 x (Time1[I_Cros1+1] – Time1[I_Cros1]) (18)
          - → Time_Cros2 = Time2[I_Cros2] + L2 x (Time2[I_Cros2+1] – Time2[I_Cros2]) (19)

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

→ Flag_Cros = 0 (20)

→ exit

- If L1 < 0 or L1 > 1 or L2 < 0 or L2 > 1, then :
  - Flag_Cros = 1 (21)

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_SEL_01 - Selection of N points for spline calculation

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	J. STUM	CLS	
Checked by:	N. PICOT	CNES	
Approved by:	P. VINCENT	CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

To select N consecutive measurements around a given point along track, to compute a spline over these N points, of a parameter versus time. Nominally, N/2 points before and N/2 points after the given point are searched. The points have to be consecutive (no time gap). If points are missing on one side, more points are considered on the other side, so as the same number of points is used for the spline calculation.

ALGORITHM SPECIFICATION

Input data

- Index of the (N/2 + 1)th measurement : Index_Higher (/)
- Minimum index of the measurement time series to consider for the search of the N points : Index_Min (/)
- Maximum index of the measurement time series to consider for the search of the N points : Index_Max (/)
- Number of measurements to be selected : N (/)
- Time tags of the measurements : Time[Index_Min:Index_Max][1]
- Time interval between two consecutive measurements : Delta_Time (s)

Output data

- Index of the first measurement to consider for the spline calculation : Index_First (/)
- Index of the last measurement to consider for the spline calculation : Index_Last (/)
- Flag for availability of the N measurements : Flag_Avail (/)[2]
- Execution status

Processing

- Flag_Avail = 0 (1)
- If N is odd, then :
  - Shift = 0 (2)
- Else :
  - Shift = 1 (3)
- M = Int(N/2) (4)

---

(1) Seconds elapsed since 01/01/1950 0h.

(2) = 0 for N/2 measurements before and N/2 measurements after available, = 1 for N measurements available but with less than N/2 measurements available on one side, = 2 for unavailable

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Checking if M points exist in the input array Time[Index_Min:Index_Max], respectively on one side before Index_Higher, and on the other side after Index_Higher. If the number of points on one side is smaller than M, then more points are considered on the other side :

- Nb_Pts_Avail_Lower = MIN(Index_Higher – Index_Min, M) (5)
- Nb_Pts_Avail_Higher = MIN(Index_Max – Index_Higher + Shift, M) (6)
- If Nb_Pts_Avail_Lower < M, then :
  - Nb_Pts_Avail_Higher = Nb_Pts_Avail_Higher + (M – Nb_Pts_Avail_Lower) (7)
  - Flag_Avail = 1 (8)
- If Nb_Pts_Avail_Higher < M, then :
  - Nb_Pts_Avail_Lower = Nb_Pts_Avail_Lower + (M – Nb_Pts_Avail_Higher) (9)
  - Flag_Avail = 1 (10)

Checking if the points on both sides are consecutive in time :

- Nb_Pts_Avail_Lower_Ini = Nb_Pts_Avail_Lower (11)
- For I = 0 to Nb_Pts_Avail_Lower_Ini – 1
  - If ABS(Time[Index_Higher – I] – Time[Index_Higher – I – 1]) > Delta_Time, then :

    Nb_Pts_Avail_Lower = I (12)

    Go out loop
- Nb_Pts_Avail_Higher_Ini = Nb_Pts_Avail_Higher (13)
- For I = 0 to Nb_Pts_Avail_Higher_Ini – 1 – Shift
  - If ABS(Time[Index_Higher + I] – Time[Index_Higher + I + 1]) > Delta_Time, then :

    Nb_Pts_Avail_Higher = I + Shift (14)

    Go out loop

If no consecutive points at all are found on one side, or of there is a time gap on both sides, then the N points are unavailable :

- If Nb_Pts_Avail_Lower = 0, or Nb_Pts_Avail_Higher = 0, or (Nb_Pts_Avail_Higher_Ini ≠ Nb_Pts_Avail_Higher and Nb_Pts_Avail_Lower_Ini ≠ Nb_Pts_Avail_Lower), then :
  - Flag_Avail = 2 (15)
  - Exit

If some points have been rejected on one single side, trying to find the same number of consecutive points on the other side :

- Nb_Pts_Avail_Higher_Fin = Nb_Pts_Avail_Higher_Ini (16)
- Nb_Pts_Avail_Lower_Fin = Nb_Pts_Avail_Lower_Ini (17)
- If Nb_Pts_Avail_Lower < Nb_Pts_Avail_Lower_Ini, then :
  - Flag_Avail = 1 (18)
  - If Nb_Pts_Avail_Higher ≥ Nb_Pts_Avail_Lower, then :

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

- For I = 1 to (Nb_Pts_Avail_Lower_Ini - Nb_Pts_Avail_Lower)
  - If {ABS(Time[Index_Higher + Nb_Pts_Avail_Higher – Shift] - Time[Index_Higher + Nb_Pts_Avail_Higher – Shift + I]) ≤ I x Delta_Time} and {Index_Higher + Nb_Pts_Avail_Higher – Shift + I ≤ Index_Max} then :
    - ⇒ Nb_Pts_Avail_Higher_Fin = Nb_Pts_Avail_Higher_Fin + 1 (19)
    - ⇒ Nb_Pts_Avail_Lower_Fin = Nb_Pts_Avail_Lower_Fin - 1 (20)
- Else :
  - Flag_Avail = 2 (21)
  - Exit

– Else :
- Flag_Avail = 2 (22)
- Exit

• If Nb_Pts_Avail_Higher < Nb_Pts_Avail_Higher_Ini, then :

– Flag_Avail = 1 (23)

– If Nb_Pts_Avail_Lower ≥ Nb_Pts_Avail_Higher, then :
- For I = 1 to (Nb_Pts_Avail_Higher_Ini - Nb_Pts_Avail_Higher)
  - If {ABS(Time[Index_Higher – Nb_Pts_Avail_Lower] - Time[Index_Higher – Nb_Pts_Avail_Lower – I]) ≤ I x Delta_Time} and {Index_Higher – Nb_Pts_Avail_Lower – I ≥ Index_Min}, then :
    - ⇒ Nb_Pts_Avail_Lower_Fin = Nb_Pts_Avail_Lower_Fin + 1 (24)
    - ⇒ Nb_Pts_Avail_Higher_Fin = Nb_Pts_Avail_Higher_Fin - 1 (25)
  - Else :
    - ⇒ Flag_Avail = 2 (26)
    - ⇒ Exit

– Else :
- Flag_Avail = 2 (27)
- Exit

• Index_First = Index_Higher – Nb_Pts_Avail_Lower_Fin (28)

• Index_Last = Index_Higher + Nb_Pts_Avail_Higher_Fin – Shift (29)

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_SEL_02 - Selection of the satellite longitude and local time for a given latitude

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	S. LABROUE	CLS	
Checked by:	N. PICOT	CNES	
Approved by:	P. VINCENT	CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

Using the satellite positions and the local time, at one second step, to select the satellite longitude and its local time, corresponding to the input latitude.

ALGORITHM SPECIFICATION

Input data

- Input latitude (in the geographic reference) : Lat_in (degrees)
- Input time-tag (seconds elapsed since 01/01/1950, in TAI) : Time_in (s)
- Satellite data at one second step :
  - Number of satellite positions : Nb_pos_sat (/)
  - For each position :
    - Time : Time_sat [0:Nb_pos_sat-1] (s)
    - Satellite latitude : Lat_sat [0:Nb_pos_sat-1] (degrees)
    - Satellite longitude : Lon_sat [0:Nb_pos_sat-1] (degrees)
    - Type of the pass : Type_sat [0:Nb_pos_sat-1] (asc or desc)
    - Local time of the satellite : Local_time_sat [0:Nb_pos_sat-1] (s)
- Maximal latitude of the satellite : Lat_sat_max (degrees)

Output data

- Selected longitude of the satellite : Lon_out (degrees)
- Selected local time : Local_time_out (s)
- Flag on the value of the input latitude : Flag_lat_max [(1)]
- Flag on the type of the pass : Flag_pass [(1)]

[(1)] 2 states: "valid" or "invalid"

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Processing

The two flags (Flag_lat_max and Flag_pass) are initialized to “invalid”.

To test the value of the input latitude :

- If (| Lat_in | > Lat_sat_max) then

  The outputs Lon_out and Local_time_out are set to a default value.
- Else :
  - The flag Flag_lat_max is set to “valid”
  - To select the starting satellite position :

    It is the satellite position given at Time_sat = Time_in which parameters are the following ones :

    Lat_sat_start, Lon_sat_start, Type_start

    If such a position is not found, the outputs Lon_out and Local_time_out are set to a default value.
  - To search the closest latitude to the input latitude (Lat_in), among the satellite latitudes Lat_sat(j).

    The latitude Lat_sat_start is used to determine the way the search must be done :
    - If (Lat_sat_start > Lat_in) then
      - If the starting position belongs to an ascending pass then

        The search is done by decrementing the indexes j
      - If the starting position belongs to a descending pass then

        The search is done by incrementing the indexes j
    - Else (Lat_sat_start < Lat_in) then
      - If the starting position belongs to an ascending pass then

        The search is done by incrementing the indexes j
      - If the starting position belongs to a descending pass then

        The search is done by decrementing the indexes j

    The found position (Lat_sat_out, Lon_sat_out, Type_out, Local_time_sat_out) must be such that it belongs to the same pass than the starting satellite position.
    - If (Type_out ≠ Type_start) then Flag_pass is “invalid” and the outputs Lon_out and Local_time_out are set to a default value.
    - If (Type_out = Type_start) then

      Lon_out = Lon_sat_out

      Local_time_out = Local_time_sat_out

      The flags Flag_pass and Flag_lat_max are set to “valid”

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

COMMENTS

None

REFERENCES

None

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

Affaires Techniques Projets et Services Opérationnels
Sous-Direction Etudes Systèmes et Développements
Division Altimétrie et Localisation Précise
Département Missions Systèmes
18, avenue Edouard Belin
31401 TOULOUSE CEDEX 4

GEN_MEC_MAT_01 - Updating a QR decomposition using Givens rotation

DEFINITION, ACCURACY AND SPECIFICATION

Prepared by:	S. LABROUE	CLS	
Checked by:	N. PICOT	CNES	
Approved by:	P. VINCENT	CNES	

Document ref:	SMM-ST-M2-EA-11010-CN	18th October, 2001	Issue: 2	Update: 4

Algorithm change record	creation	date	Issue:	Update:
	CCM			

Copyright CNES 2001. All rights reserved. No part of this document may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the prior permission of CNES.

FUNCTION

This function is used to update the QR factorization for solving a linear system.

The initial system BX = Z becomes, under the QR form, RX = T.

This function updates the matrix R and the right hand side T, applying Givens rotations to a row of the matrix B and to the corresponding component of the right hand side Z.

ALGORITHM SPECIFICATION

Input data

- Number of elements of the matrix row : N (/)
- Row of the initial matrix : Row_B[0;N-1]
- Index of the first non zero coefficient of the row : Ind_min (/)
- Component of the right hand side corresponding to the matrix row : Z_i
- Matrix R to update (triangular but stored in a vector row by row) : R[0: N*(N+1)/2 –1]
- Right hand side T to update : T[0:N-1]
- Quality index of the LS method to update : S_qual

Output data

- Matrix R updated (triangular but stored in a vector row by row) : R[0: N*(N+1)/2 –1]
- Right hand side T updated : T[0:N-1]
- Quality index of the LS method updated : S_qual

Processing

Each coefficient i of the matrix row Row_B is used to update the ith row of the matrix R and the ith component of the right hand side T, applying Givens rotations.

The rotations are applied to the elements of Row_B and to the ith row of R, updating the matrix R and extracting the information from Row_B. The parameters of the Givens rotation are computed such that the coefficient Row_B(i) is reduced to zero.

- For each coefficient i of the input row of the initial matrix, i = Ind_min, N-1
  - To compute the parameters of the Givens rotation (C and S)
    - To compute the index of the diagonal coefficient of rank i of the matrix R (as R is stored under a vector form, row by row and it is a triangular matrix)

$$Ind_diag = i * (2*N + 1 – i) /2 \quad (1)$$

- The parameters of the Givens rotation (C and S) are computed using the NAG Fortran library routine F06AAF(a,b,C,S) with the following inputs :
  - ◊ a = R(Ind_diag) (2)
  - ◊ b = Row_B(i) (3)

  The outputs are the following ones :

  - ◊ The parameters of the Givens rotation : C and S

    They are such that : $$C = \frac{a}{\sqrt{a^2 + b^2}} \quad (4)$$

    $$S = \frac{b}{\sqrt{a^2 + b^2}} \quad (5)$$

  - ◊ a is overwritten by the value d which is such that : d = C*a + S*b (6)
  - ◊ b is overwritten by the value z which is such that : z = S if |S|<C or C=0 (7)

    z = 1/C if $0 < |C| \leq S$ (8)

  The value d is used to update R(Ind_diag) :

  R(Ind_diag) = d (9)

  The ith component of Row_B is reduced o zero :

  Row_B(i) = 0 (10)

- To update the ith row of the matrix R, applying the Givens rotation to the elements j = i+1, N-1 of the matrix row Row_B and to the ith row of the matrix R

  For j = i+1, N-1

  - To compute the index of the coefficient of the matrix R

    Ind_j = Ind_diag + j – i (11)

  - To apply the Givens rotation to update R(Ind_j) (and by the way to update the matrix row Row_B)

    Temp = C * R(Ind_j) + S * Row_B(j) (12)

    Row_B(j) = -S * R(Ind_j) + C * Row_B(j) (13)

    R(Ind_j) = Temp (14)

- To update the ith component of the right hand side T, applying the Givens rotation to the input Z_i and to the ith component of T.

  Temp_bis = C * T(i) + S * Z_i (15)

  Z_i = -S * T(i) + C * Z_i (16)

  T(i) = Temp_bis (17)

- To update the quality index of the LS method

  $$S_qual = S_qual + Z_i^{\,2} \quad (18)$$

COMMENTS

None

REFERENCES

Trajectoires Spatiales, O. Zarrouati, CNES, 1987, p. 402, 403

DISTRIBUTION LIST

INTERNAL:

P. ESCUDIER	DSO/ED/AL/D	2002	
A. AURIOL	DSO/ED/AL/DO	2002	
M. COSTES			
F. PARISOT	DSO/ED/AL/JT	2002	
P. HOZE			
J. PERBOS			
M. BLUMBERGER			
G. ZAOUCHE			
S. COUTIN-FAYE	DSO/ED/AL/MA	2002	
J. NOUBEL			
P. VINCENT			
B. DELATTE			
B. BONHOURE			
N. PICOT			
G. TAVERNIER			
R. QUINTARD	DSO/ED/SG	2001	
P. SNINI			
P. GILLEN			
G. BOUTONNET	DTS/MPI/PS/SI	1501	
M. H. GOMEZ			

EXTERNAL:

M. DESTOUESSE	CLS		